دانشگاه تهران
پردیس فارابی
دانشکده‌ی مهندسی
گروه مهندسی کامپیوتر

طراحی و پیاده‌سازی سامانه مانیتورینگ و نظارت بر زیرساخت‌های فناوری اطلاعات

نگارش:

امیر میرزائی موحد

استاد راهنما:

دکتر احمد علی نژاد

استاد مشاور:

دکتر امیرحسین کیهانی پور

گزارش پروژه برای دریافت درجه‌ی کارشناسی
در رشته‌ی مهندسی کامپیوتر

دی ۱۴۰۳

بسم الله الرحمن الرحیم

چکیده

با رشد چشمگیر فناوری اطلاعات و ارتباطات، سامانه‌های نظارت و پایش به یکی از ابزارهای حیاتی برای مدیریت و بهبود عملکرد زیرساخت‌های فناوری تبدیل شده‌اند. این سامانه‌ها به کمک ابزارهای پیشرفته، داده‌های بلادرنگ و تاریخچه‌ای از شبکه‌ها، سیستم‌عامل‌ها و وب‌سایت‌ها جمع‌آوری و تحلیل می‌کنند تا مدیران را در شناسایی مشکلات و افزایش کارایی یاری دهند. در این پروژه، از Prometheus برای جمع‌آوری داده‌ها و از Grafana برای طراحی داشبوردهای گرافیکی استفاده شده است که امکان نمایش بصری و قابل‌درک اطلاعات را فراهم می‌کند.

هدف از این پروژه، طراحی و پیاده‌سازی سامانه‌ای جامع برای پایش سه حوزه اصلی شامل شبکه، سیستم‌عامل و وب‌سایت است. این سامانه با استفاده از Prometheus، داده‌های مربوط به عملکرد منابع و سرویس‌ها را جمع‌آوری کرده و از طریق داشبوردهای گرافیکی در Grafana ارائه می‌دهد. شاخص‌هایی همچون سرعت شبکه، وضعیت منابع سیستم‌عامل CPU، RAM، HDD، و زمان پاسخ‌دهی وب‌سایت‌ها به‌صورت بلادرنگ و تاریخی در این سامانه نمایش داده می‌شوند.

در فصل اول این پروژه به معرفی مساله پرداخته می‌شود. فصل دوم به بررسی ابزارها و روش‌های مانیتورینگ موجود اختصاص دارد. در فصل سوم نحوه پیاده‌سازی سامانه و استفاده از ابزارهای Prometheus و Grafana شرح داده می‌شود. فصل چهارم نتایج به‌دست‌آمده تحلیل و کارایی سامانه ارزیابی می‌شود. در نهایت، فصل پنجم به جمع‌بندی و ارائه پیشنهاداتی برای بهبود و گسترش پروژه در آینده می‌پردازد.

واژه های کلیدی : داده‌کاوی – سامانه‌های پایش – Grafana – Prometheus – شبکه – سیستم‌عامل – وب‌سایت

فهرست مطالب

۱- فصل اول - مقدمه.....۸

۱-۱-تعریف مسئله.....۹

۱-۲-اهداف.....۱۲

۱-۳-اهمیت و ضرورت.....۱۲

۲- فصل دوم - مروری بر داشبورد ها در شبکه.....۱۵

۲-۱—داشبوردهای مانیتورینگ در شبکه.....۱۶

۲-۲ — تعریف و اهمیت داشبوردهای مانیتورینگ شبکه.....۱۶

۲-۳ — ویژگی‌های داشبوردهای مانیتورینگ شبکه.....۱۶

۲-۴ — کاربردهای داشبوردهای مانیتورینگ.....۱۷

۲-۵ — مزایای استفاده از داشبوردهای مانیتورینگ.....۱۷

۲-۶—مانیتورینگ شبکه.....۱۹

۲-۶-۱ — سرعت شبکه ورودی برای همه کارت‌های شبکه.....۱۹

۲-۶-۲ — سرعت شبکه خروجی برای همه کارت‌های شبکه.....۱۹

۲-۶-۳ — تاریخچه کل ترافیک ورودی و خروجی شبکه (روزانه، ماهانه، سالانه).....۲۰

۲-۶-۴ — نمایش کانکشن‌های لحظه‌ای(مبدأ، مقصد، پروتکل و متادیتا).....۲۰

۲-۷—مانیتورینگ سیستم.....۲۱

۲-۷-۱ — خلاصه وضعیت سخت‌افزار.....۲۱

۲-۷-۲ — خلاصه وضعیت سیستم‌عامل.....۲۱

۲-۷-۳ — میزان استفاده از پردازنده (لحظه‌ای و تاریخچه).....۲۱

۲-۷-۴ — میزان استفاده از رم (لحظه‌ای و تاریخچه).....۲۲

۲-۷-۵ — میزان استفاده از هارد دیسک (لحظه‌ای و تاریخچه).....۲۲

۶-۷-۲ – لیست کاربران سیستم (کاربران فعال)..... ۲۲
۷-۷-۲ – لیست سخت‌افزارهای متصل..... ۲۳
۸-۷-۲ – لیست سرویس‌های در حال اجرا..... ۲۳
۹-۷-۲ – لاگ‌های سطح سیستم‌عامل..... ۲۳
۸-۲– مانیتورینگ وب‌سایت..... ۲۴
۱-۸-۲ – مدت‌زمان لود یک صفحه (لحظه‌ای و تاریخچه)..... ۲۴
۲-۸-۲ – نوع و نام وب‌سرور..... ۲۴
۳-۸-۲ – وضعیت آنلاین/آفلاین صفحات (لحظه‌ای و تاریخچه)..... ۲۵
۴-۸-۲ – DNS Record..... ۲۵
۵-۸-۲ – زمان پاسخ Ping..... ۲۶
۶-۸-۲ – وضعیت فعال بودن SSL و اطلاعات گواهی..... ۲۶
۹-۲ - نمونه‌های عملی از داشبوردهای مانیتورینگ شبکه..... ۲۷
۱-۹-۲ – SolarWinds Network Performance Monitor (NPM) ۲۷
۲-۹-۲ – ManageEngine OpManager..... ۲۸
۳-۹-۲ –Zabbix ۲۹
۴-۹-۲ – Datadog ۳۰
۵-۹-۲ – Monitis ۳۱
۳- فصل سوم – نصب و راه اندازی ابزار ها بر روی Ubuntu و windws..... ۳۳
۱-۳– نصب و راه اندازی ابزار ها بر روی Ubuntu..... ۳۴
۱-۱-۳- نصب و راه اندازی Prometheus ۳۴
۲-۱-۳- نصب و راه اندازی Node Exporter ۳۷
۳-۱-۳-نصب و راه اندازی curl ۳۸
۴-۱-۳- نصب و راه اندازیBlackbox Exporter..... ۴۱

۵-۱-۳- نصب و راه اندازی Grafana ۴۵
۶-۱-۳- نصب و راه اندازی Nginx ۵۱
۷-۱-۳- نصب و راه اندازی InfluxDB ۵۲
۸-۱-۳- نصب و راه اندازی Loki ۵۵
۹-۱-۳-تست صحت عملکرد..... ۵۹
۲-۳– نصب و راه اندازی ابزار ها بر روی windows..... ۶۲
۱-۲-۳ – دانلود و اجرا..... ۶۲
۲-۲-۳ – صحت اجرا..... ۶۴
۴- فصل چهارم - پیاده داشبورد ها با استفاده از Prometheus و Grafana..... ۶۸
۱-۴–مانیتورینگ شبکه..... ۶۹
۱-۱-۴ –سرعت شبکه ورودی برای همه کارت‌های شبکه..... ۶۹
۲-۱-۴ – سرعت شبکه خروجی برای همه کارت‌های شبکه..... ۷۱
۳-۱-۴ – تاریخچه کل ترافیک ورودی و خروجی شبکه ۷۳
۴-۱-۴ – نمایش کانکشن‌های لحظه‌ای..... ۷۷
۲-۴–مانیتورینگ سیستم..... ۸۱
۱-۲-۴ – خلاصه وضعیت سخت‌افزار..... ۸۱
۲-۲-۴ – خلاصه وضعیت سیستم‌عامل..... ۸۶
۳-۲-۴ – میزان استفاده از پردازنده (لحظه‌ای و تاریخچه)..... ۸۸
۴-۲-۴ – میزان استفاده از رم (لحظه‌ای و تاریخچه)..... ۹۳
۵-۲-۴ – میزان استفاده از هارد دیسک (لحظه‌ای و تاریخچه)..... ۹۹
۶-۲-۴ – تنظیم دسترسی های خاص ۱۰۶
۷-۲-۴ – لیست کاربران سیستم (کاربران فعال)..... ۱۱۴
۸-۲-۴ – لیست سخت‌افزارهای متصل..... ۱۱۷
۹-۲-۴ – لیست سرویس‌های در حال اجرا..... ۱۱۹

۱۰-۲-۴ – لاگ‌های سطح سیستم‌عامل..... ۱۲۱
۳-۴– مانیتورینگ وب‌سایت..... ۱۲۴
۱-۳-۴ – مدت‌زمان لود یک صفحه (لحظه‌ای و تاریخچه)..... ۱۲۴
۲-۳-۴ – نوع و نام وب‌سرور..... ۱۳۰
۳-۳-۴ – وضعیت آنلاین/آفلاین صفحات (لحظه‌ای و تاریخچه)..... ۱۳۲
۴-۳-۴ – DNS Record..... ۱۳۹
۵-۳-۴ – زمان پاسخ Ping..... ۱۴۴
۶-۳-۴ – وضعیت فعال بودن SSL و اطلاعات گواهی..... ۱۴۶
۵- فصل پنجم –نتیجه گیری و کارهای آینده..... ۱۵۳
۱-۵- خلاصه، جمع‌بندی و نتیجه‌گیری..... ۱۵۴
۲-۵- پیشنهادات..... ۱۵۵

فصل اول

مقدمه

این بخش با تعریف کامل مسأله آغاز شده و پس از بیان کاربردها و چالشهای آن، ساختار کلی مقاله توضیح داده شده است.

۱-۱- تعریف مسئله

در دنیای محاسبات و کامپیوتر ابزارها و نرم افزارهای نظارتی و گزارش سازی دیده بانی و ارزیابی طیف وسیعی از دانشها در علم کامپیوتر شده اند. هدف آنها بیشتر پایش و بررسی عملکرد بازدهی وضعیت سلامت کارایی ابزارها و نرم افزارها می باشند.

سامانه های نظارتی در دنیای محاسبات به دسته های زیر تقسیم می شوند

- مدیریت عملکرد برنامه
- نظارت بر رویداد
- مدیریت معاملات تجاری
- مانیتورینگ شبکه
- نظارت بر سیستم
- نظارت بر وب سایت
- نظارت بر فعالیت کاربر

هر کدام از دسته های بالا خود شامل نرم افزارها و فرم ورکهای کوچک و بزرگ رایگان و انترپرایز هستند.

در طول سالیانه گذشته رویکردهای مختلفی در آنالیز و مانیتورینگ در حوزه کامپیوتر مورد توجه قرار گرفته و این موضوع به خصوص برای سامانه های ریل تایم دیتاسنترها سامانه پشتیبانی سیستمهای نظارت و حراست و .... بسیار پرکاربرد بوده است.

همیشه وجود یک سیستم مانیتور که به شما در لحاظ حساس که خودتان آنها را قبلا توسط یکسری نقاط استانه تعریف کرده اید هشدار لازم را ارسال کند در نگه داری به مدیران سیستم ها و مدیران حوزه مانیتورینگ کمکهای شایانی می کند.

ابزارهای متن باز و فریور و انترپرایز در این حوزه تعداد زیادی هستند که مورد توجه کاربران قرار گرفته اند.

در این مساله ما بر سه مورد از ۷ مورد از دسته های ابزارهای مانیتورینگ تمرکز می کنیم

۱. مانیتورینگ شبکه
۲. سیستم عامل
۳. وب سایت

۱. مانیتورینگ شبکه

۱. سرعت شبکه ورودی برای همه کارتهای شبکه
۲. سرعت شبکه خروجی برای همه کارتهای شبکه
۳. هیستوری از کل ترافیک ورودی ، خروجی شبکه در دوره های روزانه ماهانه سالانه .
٤. نمایش کانکشنهای در لحظه مبدا مقصد پرت یا پرتکل متادیتا)

۲. مانیتورینگ سیستم

۱. . خلاصه وضعیت سخت افزار

۲. . خلاصه وضعیت سیستم عامل

۳. میزان استفاده از پردازنده لحظه ای تاریخچه ۵،۱۰،۱۵ دقیقه

٤. . میزان استفاده از رم لحظه ای تاریخچه ۵،۱۰،۱۵ دقیقه

٥. میزان استفاده از هارد لحظه ای تاریخچه ۵،۱۰،۱۵ دقیقه

٦. لیست کاربران سیستم کاربران فعال

۷. لیست سخت افزارهای متصل

۸. لیست سرویسهای در حال اجرا

۹. لاگهای سطح سیستم عامل

۳. .مانیتورینگ وب سایت

۱. مدت زمان لود یک صفحه لحظه ای تاریخچه ۵،۱۰،۱۵ دقیقه)

۲. نوع و نام وب سرور

۳. وضعیت آنلاین آفلاین صفحه لحظه ای تاریخچه روزانه ماهانه سالانه)

٤. DNS Record

٥. Ping time

٦. وضعیت فعال بودن ssl مدت زمان گواهی و اطلاعات گواهی

۲-۱- اهداف

هدف از انجام این پروژه طراحی و پیاده‌سازی یک سامانه جامع مانیتورینگ برای سازمان‌ها است که بتواند سه حوزه اصلی شامل شبکه، سیستم‌عامل و وب‌سایت‌ها را پایش کند. این سامانه با استفاده از ابزار قدرتمند Prometheus، داده‌های مربوط به عملکرد منابع و سرویس‌ها را جمع‌آوری کرده و از طریق Grafana به‌صورت گرافیکی و بصری ارائه می‌دهد.

در این پروژه، تمرکز بر جمع‌آوری و تحلیل داده‌های بلادرنگ و تاریخی است تا شاخص‌هایی همچون سرعت و وضعیت شبکه، منابع سیستم‌عامل (CPU، RAM، HDD) و زمان پاسخ‌دهی وب‌سایت‌ها شناسایی و پایش شوند.

این سامانه به مدیران سازمان کمک می‌کند تا با کشف الگوهای عملکردی و شناسایی مشکلات احتمالی، بهره‌وری زیرساخت‌های فناوری را بهبود بخشیده و تصمیم‌گیری‌های بهتری انجام دهند. حوزه کاربرد این پروژه شامل مدیریت بهتر منابع فناوری، افزایش کارایی زیرساخت‌ها، و شناسایی سریع نواقص در شبکه و سیستم‌های وابسته است.

۳-۱- اهمیت و ضرورت

در دنیای امروز، با افزایش وابستگی سازمان‌ها به فناوری اطلاعات و ارتباطات، پایش و مدیریت مؤثر زیرساخت‌های فناوری به یکی از اولویت‌های اصلی تبدیل شده است. نظارت بر شبکه، سیستم‌عامل‌ها و وب‌سایت‌ها، نه‌تنها برای شناسایی سریع مشکلات، بلکه برای بهبود عملکرد و بهره‌وری نیز ضروری است.

استفاده از ابزارهای پیشرفته‌ای مانند Prometheus و Grafana، امکان جمع‌آوری داده‌های دقیق و ارائه آن‌ها به‌صورت بصری را فراهم می‌کند که این داده‌ها نقش کلیدی در تصمیم‌گیری‌های مدیریتی دارند.

طراحی یک سامانه مانیتورینگ جامع، این امکان را فراهم می‌آورد که سازمان‌ها بتوانند مشکلات زیرساختی را به‌صورت بلادرنگ شناسایی کرده و اقدامات پیشگیرانه را در زمان مناسب انجام دهند. این امر نه‌تنها باعث صرفه‌جویی در هزینه‌ها و منابع می‌شود، بلکه بهره‌وری و اعتماد به خدمات فناوری را نیز افزایش می‌دهد.

با توجه به اهمیت افزایش سرعت پاسخ‌گویی، کاهش خرابی‌ها، و بهبود تجربه کاربران، پیاده‌سازی این پروژه برای سازمان‌ها امری حیاتی به شمار می‌رود. چنین سامانه‌ای می‌تواند سازمان‌ها را در رقابت‌های پیچیده بازار مدرن به سمت موفقیت و پایداری سوق دهد.

فصل دوم

مروری بر داشبورد ها در شبکه

۲-۱- داشبوردهای مانیتورینگ در شبکه

داشبوردهای مانیتورینگ شبکه ابزارهایی پیشرفته هستند که به مدیران فناوری اطلاعات امکان مشاهده و مدیریت عملکرد شبکه را در قالبی ساده و گرافیکی می‌دهند. این داشبوردها با نمایش اطلاعات به‌صورت بلادرنگ و تجزیه و تحلیل داده‌های گذشته، ابزار مهمی برای مدیریت شبکه‌های مدرن محسوب می‌شوند. در این متن به تعریف، ویژگی‌ها، و مزایای داشبوردهای مانیتورینگ پرداخته می‌شود.

۲-۲- تعریف و اهمیت داشبوردهای مانیتورینگ شبکه

داشبورد شبکه نمایی بصری از وضعیت فعلی و عملکرد شبکه ارائه می‌دهد. این ابزارها امکان بررسی اجزای مختلف شبکه از جمله روترها، سوئیچ‌ها، سرورها و دستگاه‌های متصل را فراهم کرده و داده‌های مربوط به ترافیک، پهنای باند و امنیت را نمایش می‌دهند. با استفاده از این اطلاعات، مدیران می‌توانند مشکلات را در لحظه شناسایی کرده و برای رفع آن‌ها اقدام کنند.

۲-۳- ویژگی‌های داشبوردهای مانیتورینگ شبکه

۱. **انعطاف‌پذیری و سفارشی‌سازی:** داشبوردهای مانیتورینگ ابزارهایی انعطاف‌پذیر و قابل سفارشی‌سازی هستند. هر سازمان بسته به نیازهای خود می‌تواند بخش‌های مختلف داشبورد را تنظیم کند تا اطلاعات خاصی را مشاهده نماید. این اطلاعات به‌صورت نمودارها، گراف‌ها و جداول نمایش داده می‌شوند و فهم آن‌ها حتی برای افرادی که تخصص زیادی در فناوری اطلاعات ندارند، ساده است.

۲. **اطلاعات بلادرنگ:** یکی از مهم‌ترین مزایای داشبوردهای مانیتورینگ، ارائه اطلاعات بلادرنگ است. در صورتی که مشکلی در شبکه رخ دهد، داشبورد بلافاصله هشدارهای لازم را برای مدیران ارسال می‌کند. این هشدارها می‌توانند از طریق ایمیل، پیامک یا سایر روش‌های ارتباطی ارسال شوند.

۳. **ذخیره‌سازی تاریخچه:** داشبوردها اطلاعات مربوط به تاریخچه شبکه را نیز ذخیره می‌کنند که برای تحلیل روندها و پیش‌بینی نیازهای آینده بسیار مفید است.

۲-۴- کاربردهای داشبوردهای مانیتورینگ

داشبوردهای مانیتورینگ به بهبود امنیت شبکه کمک شایانی می‌کنند. آن‌ها با شناسایی رفتارهای غیرعادی یا دسترسی‌های غیرمجاز، امکان واکنش سریع به تهدیدات امنیتی را فراهم می‌آورند. همچنین، این ابزارها برای انطباق با استانداردهای امنیتی و ارائه گزارش‌های لازم به مقامات نظارتی بسیار مفید هستند.

مدیران می‌توانند با استفاده از اطلاعات ارائه‌شده توسط داشبوردها، بهره‌وری اجزای شبکه را افزایش دهند و از منابع سخت‌افزاری به شکل بهینه‌تری استفاده کنند. این امر باعث کاهش هزینه‌های عملیاتی و افزایش کارایی کلی سیستم می‌شود.

داشبوردها امکان مدیریت و نظارت بر تمامی بخش‌های شبکه را در یک صفحه واحد فراهم می‌کنند و اطلاعات مختلفی مانند استفاده از پهنای باند، ترافیک، و وضعیت سرورها را نمایش می‌دهند.

۲-۵- مزایای استفاده از داشبوردهای مانیتورینگ

۱. **صرفه‌جویی در زمان :**مشاهده سریع وضعیت شبکه بدون نیاز به بررسی دستی داده‌ها.
۲. **کاهش هزینه‌ها :**جلوگیری از خرابی‌های پرهزینه با شناسایی سریع مشکلات.
۳. **افزایش قابلیت اعتماد :**مانیتورینگ مداوم به کاهش زمان خرابی و افزایش اعتماد به سیستم کمک می‌کند.
۴. **بهبود تصمیم‌گیری :**ارائه داده‌های دقیق و قابل تحلیل برای تصمیم‌گیری‌های مدیریتی.

به طور کلی، داشبوردهای مانیتورینگ شبکه ابزارهایی هستند که با ارائه اطلاعات دقیق و کاربردی، به سازمان‌ها کمک می‌کنند تا بهره‌وری، امنیت و پایداری شبکه‌های خود را افزایش دهند. استفاده از این داشبوردها در سازمان‌های امروزی، به‌ویژه سازمان‌هایی که شبکه‌های پیچیده و گسترده دارند، ضروری است. این ابزارها نه‌تنها فرآیند مدیریت شبکه را ساده‌تر می‌کنند، بلکه به مدیران این امکان را می‌دهند تا به جای صرف زمان برای شناسایی مشکلات، بر روی بهبود و توسعه شبکه تمرکز کنند.

حال قصد داریم داشبوردهای مانیتورینگ را در سه دسته اصلی پیاده‌سازی کنیم که هرکدام به جنبه‌های مختلف عملکرد سیستم و شبکه توجه دارند. این دسته‌ها عبارتند از: مانیتورینگ شبکه، سیستم‌عامل، و وب‌سایت. هر یک از این دسته‌ها با تمرکز بر جنبه‌های خاصی از زیرساخت‌های فناوری اطلاعات، به مدیران سیستم این امکان را می‌دهد که وضعیت، عملکرد، و امنیت را به طور دقیق نظارت کنند.

دسته اول، مانیتورینگ شبکه، به تحلیل و نظارت بر وضعیت ارتباطات و عملکرد شبکه می‌پردازد. در این دسته، اطلاعاتی مانند زمان پاسخ Ping، وضعیت اتصال به شبکه، و سرعت انتقال داده‌ها بررسی می‌شود. دسته دوم، مانیتورینگ سیستم‌عامل، به وضعیت و عملکرد سیستم‌عامل‌های مختلف سرورها و دستگاه‌ها می‌پردازد و اطلاعاتی مانند منابع سیستمی، وضعیت سرویس‌ها، و فعالیت‌های کاربران را شامل می‌شود. دسته سوم، مانیتورینگ وب‌سایت، بر نظارت و بهینه‌سازی عملکرد وب‌سایت‌ها تمرکز دارد و اطلاعاتی نظیر زمان لود صفحات، وضعیت آنلاین/آفلاین بودن، و تنظیمات DNS را فراهم می‌آورد. در ادامه، با بررسی جزئیات ۱۹ داشبورد این سه دسته، نحوه پیاده‌سازی و کاربرد هر یک از این داشبوردها را توضیح خواهیم داد.

۶-۲- مانیتورینگ شبکه

۱-۶-۲- سرعت شبکه ورودی برای همه کارت‌های شبکه

این داشبورد به‌طور خاص برای نمایش میزان سرعت شبکه ورودی برای تمامی کارت‌های شبکه سیستم طراحی شده است. اطلاعات ارائه‌شده در این داشبورد، شامل میزان دیتایی است که از منابع مختلف وارد سیستم می‌شود. این ابزار به مدیران شبکه کمک می‌کند تا در لحظه از وضعیت دریافت داده‌های ورودی مطلع شوند و عملکرد کارت‌های شبکه را بررسی کنند. اهمیت این داشبورد در مدیریت ترافیک شبکه و شناسایی مشکلاتی مانند اختلالات احتمالی در اتصال، حملات DDoS یا کمبود پهنای باند نمایان می‌شود. همچنین، با مانیتورینگ سرعت ورودی، می‌توان الگوهای رفتاری شبکه را شناسایی و اقدامات لازم برای بهبود عملکرد سیستم را انجام داد. این داشبورد نقش کلیدی در حفظ پایداری و امنیت شبکه دارد.

۲-۶-۲- سرعت شبکه خروجی برای همه کارت‌های شبکه

این داشبورد میزان داده‌هایی را که از سیستم به سایر منابع ارسال می‌شود، نمایش می‌دهد. اطلاعات مربوط به سرعت شبکه خروجی برای تمامی کارت‌های شبکه در اینجا ثبت و مانیتور می‌شود. کاربرد اصلی این داشبورد در تحلیل عملکرد سیستم، اطمینان از انتقال صحیح داده‌ها و جلوگیری از مشکلاتی مانند اشباع پهنای باند یا ارسال داده‌های غیرضروری است. به کمک این ابزار، مدیران می‌توانند بهینه‌سازی سیستم را انجام داده و از عملکرد صحیح ارتباطات خروجی اطمینان حاصل کنند. اهمیت این داشبورد به‌ویژه در مواقعی که حجم بالایی از داده‌ها ارسال می‌شود، مشخص می‌گردد، زیرا می‌تواند کمک کند که منابع به‌درستی مدیریت شوند.

۳-۶-۲- تاریخچه کل ترافیک ورودی و خروجی شبکه (روزانه، ماهانه، سالانه)

این داشبورد تاریخچه جامعی از ترافیک ورودی و خروجی شبکه در بازه‌های زمانی مختلف ارائه می‌دهد. اطلاعات جمع‌آوری‌شده شامل مجموع داده‌های ارسال و دریافت‌شده در طول روز، ماه و سال است. این ابزار برای تحلیل روند مصرف شبکه و برنامه‌ریزی بلندمدت بسیار مفید است. با استفاده از این داشبورد، می‌توان روندهای غیرعادی را شناسایی و به موقع به مشکلات واکنش نشان داد. اهمیت این ابزار در برنامه‌ریزی برای ارتقاء زیرساخت‌های شبکه و پیشگیری از مشکلاتی مانند کمبود پهنای باند یا حملات سایبری برجسته می‌شود. همچنین این داشبورد می‌تواند به مدیران شبکه کمک کند که تصمیمات استراتژیک‌تری برای مدیریت منابع بگیرند.

۴-۶-۲- نمایش کانکشن‌های لحظه‌ای (مبدأ، مقصد، پروتکل و متادیتا)

این داشبورد یکی از کاربردی‌ترین ابزارها برای تحلیل وضعیت لحظه‌ای شبکه است. با استفاده از این داشبورد، می‌توان تمامی ارتباطات جاری در سیستم را مشاهده کرد. اطلاعاتی نظیر آدرس مبدأ و مقصد، شماره پورت، نوع پروتکل و سایر متادیتاها نمایش داده می‌شود. اهمیت این داشبورد در شناسایی ارتباطات مشکوک، پیشگیری از حملات سایبری و بهبود مدیریت ترافیک شبکه نهفته است. این ابزار همچنین به مدیران شبکه کمک می‌کند تا ارتباطات غیرمجاز یا نامناسب را شناسایی کرده و اقدامات لازم را برای مسدود کردن آن‌ها انجام دهند. علاوه بر این، داشبورد مذکور یک نمای کلی از سلامت و کارایی شبکه ارائه می‌دهد که برای نظارت دقیق ضروری است.

۷-۲- مانیتورینگ سیستم

۱-۷-۲- خلاصه وضعیت سخت‌افزار

این داشبورد اطلاعات جامعی از وضعیت سخت‌افزار سیستم ارائه می‌دهد. داده‌هایی مانند دما، ولتاژ، سرعت فن، سلامت هارد دیسک و سایر جزئیات مرتبط با قطعات مختلف سخت‌افزاری نمایش داده می‌شود. این داشبورد به مدیران سیستم کمک می‌کند تا عملکرد سخت‌افزار را به‌صورت بلادرنگ و همچنین در بازه‌های زمانی مختلف نظارت کنند. اهمیت این ابزار در شناسایی مشکلات سخت‌افزاری پیش از وقوع خرابی جدی است. با استفاده از این اطلاعات، می‌توان از آسیب‌های احتمالی به سیستم جلوگیری کرد و برای تعمیر یا ارتقاء سخت‌افزار برنامه‌ریزی کرد. همچنین، این داشبورد نقش کلیدی در تضمین پایداری عملکرد سیستم‌های حیاتی دارد.

۲-۷-۲- خلاصه وضعیت سیستم‌عامل

این داشبورد یک نمای کلی از وضعیت سیستم‌عامل، شامل نسخه، به‌روزرسانی‌ها، فرآیندهای فعال و سلامت کلی آن ارائه می‌کند. اطلاعات ارائه‌شده در این داشبورد برای مدیران سیستم بسیار مفید است، زیرا امکان شناسایی مشکلات نرم‌افزاری و تحلیل عملکرد سیستم‌عامل را فراهم می‌کند. با استفاده از این ابزار، می‌توان منابع سیستم را بهینه‌سازی کرد و مشکلات احتمالی نظیر بار اضافی روی پردازنده یا رم را کاهش داد. اهمیت این داشبورد در مدیریت بهینه سیستم و افزایش کارایی کلی آن به‌وضوح قابل مشاهده است.

۳-۷-۲- میزان استفاده از پردازنده (لحظه‌ای و تاریخچه)

این داشبورد اطلاعات مربوط به میزان استفاده از پردازنده را در لحظه و در بازه‌های زمانی ۵، ۱۰ و ۱۵ دقیقه ارائه می‌دهد. با استفاده از این ابزار، مدیران سیستم می‌توانند فشار کاری روی پردازنده را مانیتور کرده و از وجود

مشکلاتی نظیر بار اضافی یا کاهش کارایی جلوگیری کنند. این داشبورد به‌ویژه در محیط‌های کاری پرمشغله و سیستم‌هایی با بار کاری بالا اهمیت دارد، زیرا می‌تواند نشانه‌های اولیه مشکلات عملکردی را آشکار کرده و از خرابی سیستم جلوگیری کند. اطلاعات تاریخچه نیز به تحلیل روند مصرف و پیش‌بینی نیازهای آینده کمک می‌کند.

۲-۷-۴- میزان استفاده از رم (لحظه‌ای و تاریخچه)

این داشبورد میزان استفاده از حافظه رم را به‌صورت لحظه‌ای و در بازه‌های زمانی کوتاه‌مدت نمایش می‌دهد. اطلاعات ارائه‌شده شامل مصرف حافظه توسط فرآیندهای فعال و تاریخچه آن در ۵، ۱۰ و ۱۵ دقیقه گذشته است. این داشبورد برای شناسایی فرآیندهایی که بیش از حد حافظه مصرف می‌کنند، بسیار مفید است. اهمیت این ابزار در جلوگیری از کمبود منابع حافظه، بهبود کارایی سیستم و ارائه راهکارهایی برای مدیریت بهتر منابع قابل توجه است.

۲-۷-۵- میزان استفاده از هارد دیسک (لحظه‌ای و تاریخچه)

این داشبورد اطلاعاتی در مورد میزان استفاده از هارد دیسک سیستم، شامل فضاهای اشغال‌شده و آزاد را ارائه می‌دهد. همچنین، تاریخچه مصرف هارد در بازه‌های زمانی کوتاه‌مدت به نمایش درمی‌آید. این اطلاعات برای تحلیل روند مصرف دیسک و جلوگیری از مشکلاتی مانند پر شدن حافظه ذخیره‌سازی یا کاهش سرعت عملکرد سیستم بسیار اهمیت دارد. این داشبورد به مدیران کمک می‌کند تا بهینه‌سازی منابع ذخیره‌سازی را به‌صورت دقیق‌تر انجام دهند.

۲-۷-۶- لیست کاربران سیستم (کاربران فعال)

این داشبورد فهرستی از کاربران سیستم و وضعیت فعالیت آن‌ها را نمایش می‌دهد. اطلاعات ارائه‌شده شامل نام کاربران، زمان ورود و وضعیت فعالیت آن‌ها است. این ابزار به مدیران سیستم کمک می‌کند تا نظارت دقیقی بر

فعالیت کاربران داشته باشند و از امنیت سیستم اطمینان حاصل کنند. همچنین، شناسایی کاربران مشکوک یا فعالیت‌های غیرمجاز از دیگر مزایای این داشبورد است.

۷-۷-۲- لیست سخت‌افزارهای متصل

این داشبورد اطلاعات مربوط به تمامی سخت‌افزارهای متصل به سیستم، شامل نوع، وضعیت و مشخصات آن‌ها را نمایش می‌دهد. اطلاعات ارائه‌شده برای مدیریت و نگهداری از سخت‌افزارها، شناسایی مشکلات و برنامه‌ریزی برای ارتقاء سیستم بسیار مفید است. این داشبورد به شناسایی سخت‌افزارهای ناکارآمد یا ناسازگار کمک کرده و نقش کلیدی در بهبود عملکرد کلی سیستم دارد.

۸-۷-۲- لیست سرویس‌های در حال اجرا

این داشبورد تمامی سرویس‌های در حال اجرا در سیستم را همراه با جزئیاتی نظیر وضعیت، منابع مصرفی و مدت‌زمان فعالیت آن‌ها نمایش می‌دهد. این اطلاعات برای تحلیل عملکرد سیستم، شناسایی سرویس‌های ناکارآمد و بهینه‌سازی منابع بسیار اهمیت دارد. همچنین، این ابزار به مدیران کمک می‌کند تا از فعال بودن سرویس‌های حیاتی اطمینان حاصل کنند.

۹-۷-۲- لاگ‌های سطح سیستم‌عامل

این داشبورد لاگ‌های سیستم‌عامل، شامل پیام‌های خطا، هشدارها و فعالیت‌های اخیر را نمایش می‌دهد. این اطلاعات به مدیران سیستم کمک می‌کند تا مشکلات نرم‌افزاری و سخت‌افزاری را شناسایی کرده و برای رفع آن‌ها اقدام کنند. این داشبورد به‌ویژه برای تحلیل رخدادهای امنیتی و بهبود عملکرد کلی سیستم اهمیت زیادی دارد.

۸-۲- مانیتورینگ وب‌سایت

۱-۸-۲- مدت‌زمان لود یک صفحه (لحظه‌ای و تاریخچه)

یکی از شاخص‌های کلیدی عملکرد وب‌سایت، مدت‌زمان لود صفحات است. در داشبورد مدت‌زمان لود، اطلاعات لحظه‌ای و تاریخی نمایش داده می‌شود که نشان‌دهنده زمان صرف‌شده برای بارگذاری یک صفحه در حال حاضر و در بازه‌های زمانی ۵، ۱۰ و ۱۵ دقیقه گذشته است. این داده‌ها به مدیران وب‌سایت کمک می‌کند تا عملکرد سایت را نظارت کرده و در صورت لزوم اقدامات بهینه‌سازی انجام دهند. مدت‌زمان بارگذاری صفحات ارتباط مستقیمی با تجربه کاربری دارد؛ هرچه زمان بارگذاری طولانی‌تر باشد، احتمال ترک سایت توسط کاربران بیشتر می‌شود. بهینه‌سازی سرعت سایت می‌تواند با استفاده از ابزارهایی مانند فشرده‌سازی تصاویر، کش‌گذاری صحیح، و کاهش درخواست‌های HTTP بهبود یابد. علاوه بر این، این داشبورد می‌تواند تغییرات در سرعت بارگذاری صفحات را در طول زمان تحلیل کرده و به مدیران کمک کند که مشکلات احتمالی را شناسایی و برطرف کنند.

۲-۸-۲- نوع و نام وب‌سرور

نوع وب‌سرور و نسخه آن اطلاعات مهمی برای مدیریت سایت و تحلیل‌های نرم‌افزاری به شمار می‌آید. در این داشبورد، اطلاعاتی در مورد وب‌سرور سایت، مانند نوع و نسخه آن، نمایش داده می‌شود. آگاهی از این اطلاعات به مدیران کمک می‌کند تا از سازگاری نرم‌افزارهای مختلف با وب‌سرور اطمینان حاصل کنند. علاوه بر این، وب‌سرور نقش مهمی در عملکرد و امنیت سایت دارد. شناخت دقیق نوع وب‌سرور و نسخه آن می‌تواند به شناسایی مشکلات موجود و اجرای به‌روزرسانی‌های لازم کمک کند. همچنین، این داشبورد برای ارتقاء امنیت سایت نیز اهمیت دارد؛ زیرا با تشخیص آسیب‌پذیری‌های احتمالی در نسخه‌های قدیمی وب‌سرور، می‌توان اقدامات پیشگیرانه‌ای برای کاهش خطرات امنیتی انجام داد. به عنوان مثال، استفاده از وب‌سرورهای قدیمی ممکن است باعث بروز مشکلاتی در سازگاری با استانداردهای امنیتی جدید و همچنین آسیب‌پذیری در برابر حملات شود.

۳-۸-۲- وضعیت آنلاین/آفلاین صفحات (لحظه‌ای و تاریخچه)

وضعیت آنلاین/آفلاین صفحات به مدیران وب‌سایت این امکان را می‌دهد که وضعیت دسترسی به صفحات سایت را در لحظه و در بازه‌های زمانی مختلف (روزانه، ماهانه، سالانه) بررسی کنند. این داشبورد نشان می‌دهد که آیا صفحات سایت در دسترس هستند یا خیر و مدت‌زمانی که صفحات سایت به دلایلی مانند مشکلات سرور یا اتصال به اینترنت آفلاین بوده‌اند، چیست. بررسی وضعیت آنلاین/آفلاین صفحات در طول زمان می‌تواند به شناسایی الگوهای مشکلات کمک کند و امکان رفع مشکلات به‌موقع را فراهم کند. علاوه بر این، اطلاعات مربوط به قطعی‌ها و مدت‌زمان آفلاین بودن می‌تواند به تحلیل عملکرد و بهبود پایداری سیستم کمک کند. برای مثال، اگر سایت به طور مکرر در زمان‌های خاصی آفلاین شود، این اطلاعات می‌تواند به شناسایی مشکلات زیرساختی کمک کند که نیاز به تعمیرات یا به‌روزرسانی دارند.

۴-۸-۲- DNS Record

رکوردهای DNS، اطلاعات مهمی هستند که نحوه دسترسی به سایت را تعیین می‌کنند. این داشبورد اطلاعات مربوط به رکوردهای DNS سایت را نمایش می‌دهد و به مدیران سایت کمک می‌کند تا از تنظیمات صحیح DNS اطمینان حاصل کنند. یکی از مشکلات رایج در ارتباطات اینترنتی، مشکلات مربوط به DNS است که می‌تواند باعث عدم دسترسی به سایت یا کندی بارگذاری صفحات شود. بنابراین، نظارت دقیق بر رکوردهای DNS برای تشخیص مشکلات و رفع آن‌ها بسیار مهم است. این داشبورد به مدیران امکان می‌دهد تا به راحتی تنظیمات DNS را بررسی کنند و از صحت آن‌ها مطمئن شوند. به عنوان مثال، اطمینان از اینکه رکوردهای A، CNAME و MX به درستی تنظیم شده‌اند می‌تواند به پیشگیری از مشکلات دسترسی به سایت و ارسال ایمیل‌ها کمک کند.

۵-۸-۲- زمان پاسخ Ping

زمان پاسخ Ping یکی از مهم‌ترین شاخص‌ها برای ارزیابی کیفیت ارتباط شبکه است. این داشبورد زمان پاسخ سایت به درخواست‌های Ping را نمایش می‌دهد که نشان‌دهنده زمان تاخیر بین ارسال درخواست و دریافت پاسخ از سرور است. این داده‌ها برای مدیران شبکه و وب‌سایت‌ها بسیار مفید هستند، زیرا با تجزیه و تحلیل زمان پاسخ Ping می‌توانند مشکلات احتمالی در ارتباطات شبکه را شناسایی کنند. افزایش زمان پاسخ می‌تواند نشان‌دهنده مشکلاتی مانند ترافیک شبکه بالا، مشکلات سرور، یا خرابی در مسیر ارتباطی باشد. این داشبورد به مدیران این امکان را می‌دهد که به سرعت مشکلات مربوط به سرعت شبکه را شناسایی کرده و اقدامات لازم را برای بهبود عملکرد سایت انجام دهند.

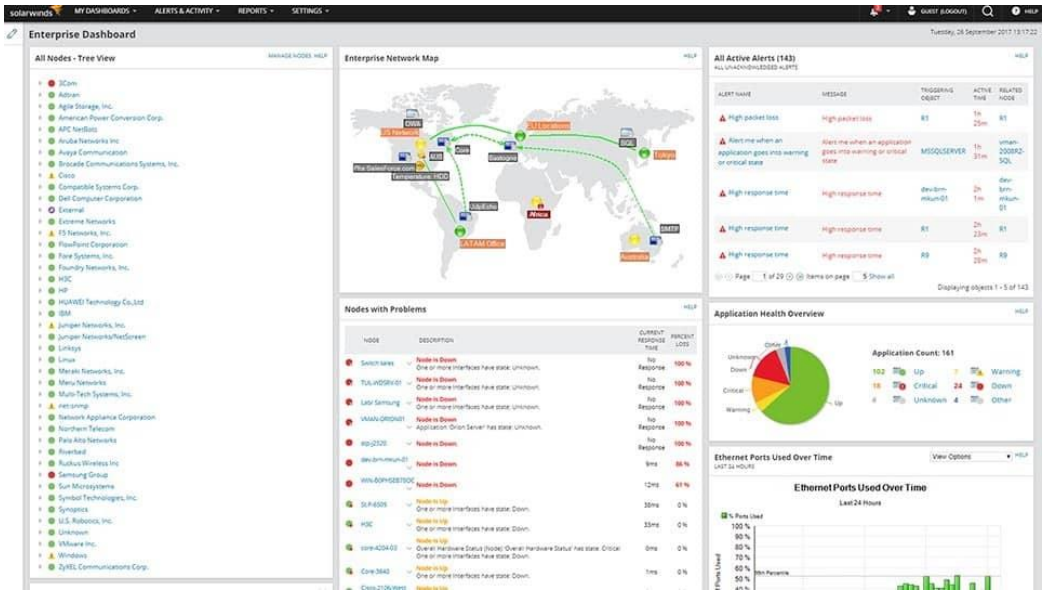
۶-۸-۲- وضعیت فعال بودن SSL و اطلاعات گواهی

وضعیت SSL و اطلاعات گواهی سایت از جنبه‌های حیاتی امنیت وب‌سایت هستند. این داشبورد وضعیت فعال بودن SSL، مدت‌زمان باقی‌مانده تا انقضای گواهی و جزئیات مربوط به آن را نمایش می‌دهد. SSL (Secure Sockets Layer) برای رمزنگاری اطلاعات بین مرورگر کاربر و سرور وب‌سایت استفاده می‌شود و به محافظت از داده‌های حساس مانند اطلاعات کارت اعتباری و اطلاعات شخصی کمک می‌کند. این داشبورد به مدیران سایت کمک می‌کند تا از فعال بودن SSL و تاریخ انقضای گواهی‌ها مطلع شوند و در صورت نیاز، گواهی‌های SSL خود را به‌روزرسانی کنند. همچنین، این ابزار به مدیران امکان می‌دهد تا مشکلات احتمالی مرتبط با SSL را شناسایی کرده و اقدامات لازم برای حفظ امنیت سایت را انجام دهند. به طور کلی، بررسی وضعیت SSL و گواهی‌ها برای جلوگیری از حملات سایبری و محافظت از داده‌ها اهمیت زیادی دارد.

۹-۲-نمونه‌های عملی از داشبوردهای مانیتورینگ شبکه

۱-۹-۲- SolarWinds Network Performance Monitor (NPM) :

SolarWinds NPM یکی از قدرتمندترین ابزارهای مانیتورینگ شبکه است که با داشبوردهای قابل تنظیم خود، امکان نظارت بر سلامت شبکه، پهنای باند و عملکرد دستگاه‌های متصل را فراهم می‌کند. این ابزار برای سازمان‌هایی با زیرساخت‌های پیچیده بسیار مناسب است و اطلاعات بلادرنگ و جامعی را ارائه می‌دهد.

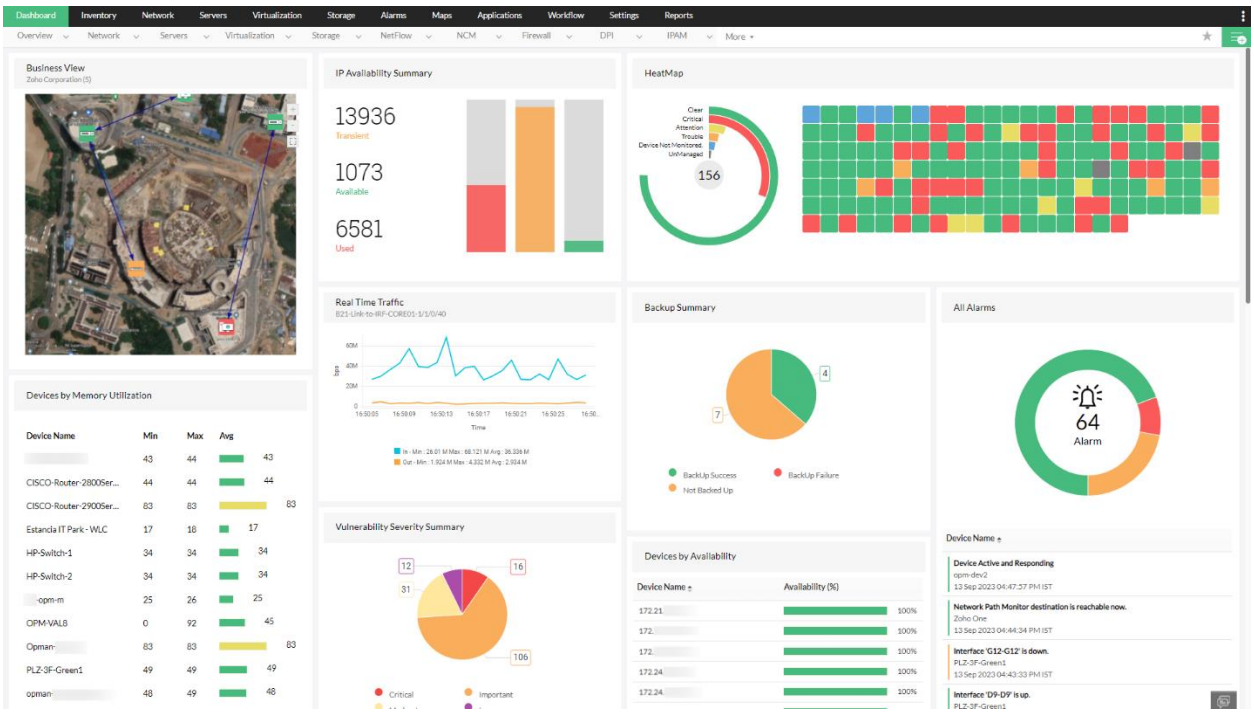


شکل ۱-۲: داشبورد SolarWinds NPM با نمایش وضعیت سلامت شبکه و دستگاه‌های متصل.

این ابزار با داشبوردهای قابل تنظیم، امکان نظارت بر سلامت شبکه، پهنای باند و عملکرد دستگاه‌های متصل را فراهم می‌کند .

۲-۹-۲- ManageEngine OpManager :

ManageEngine OpManager یکی دیگر از ابزارهای پیشرفته مانیتورینگ شبکه است. این نرم‌افزار با داشبوردهای گرافیکی و کاربرپسند، اطلاعاتی مانند استفاده از CPU، پهنای باند، و فضای دیسک را نمایش می‌دهد. این ابزار برای سازمان‌هایی که نیاز به مدیریت منابع شبکه دارند، بسیار کاربردی است.



شکل ۲-۱: داشبورد ManageEngine OpManager با قابلیت نمایش جزئیات منابع سخت‌افزاری.

این نرم‌افزار با ارائه داشبوردهای گرافیکی، نظارت بر سرورها، روترها و سوئیچ‌ها را تسهیل کرده و اطلاعاتی مانند استفاده از CPU و فضای دیسک را نمایش می‌دهد .

۳-۹-۲- Zabbix :

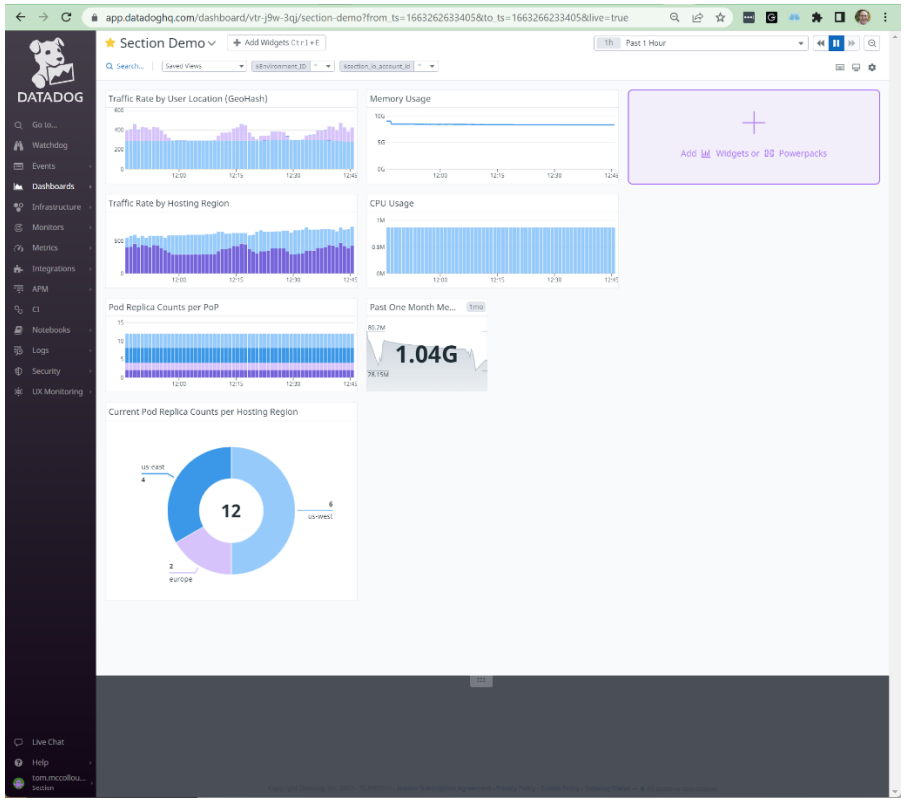
Zabbix یک ابزار متن‌باز و رایگان برای مانیتورینگ شبکه است. این نرم‌افزار امکان نظارت بر عملکرد سرورها، شبکه‌ها و برنامه‌های کاربردی را فراهم می‌کند. داشبوردهای Zabbix به‌صورت انعطاف‌پذیر قابل تنظیم هستند و می‌توانند برای سازمان‌های کوچک و بزرگ استفاده شوند.

شکل ۳-۲: داشبورد Zabbix با قابلیت‌های مانیتورینگ متن‌باز و گرافیکی.

یک ابزار متن‌باز با داشبوردهای جامع که امکان نظارت بر عملکرد سرورها، شبکه‌ها و برنامه‌های کاربردی را به‌صورت بلادرنگ فراهم می‌کند .

۴-۹-۲- Datadog :

Datadog ابزاری ابری برای مانیتورینگ شبکه، سرورها و برنامه‌ها است. این ابزار با داشبوردهای پیشرفته، امکان نظارت بلادرنگ را فراهم کرده و با بیش از ۴۰۰ ابزار دیگر یکپارچه می‌شود. Datadog برای سازمان‌هایی که از خدمات ابری استفاده می‌کنند، بسیار مناسب است.

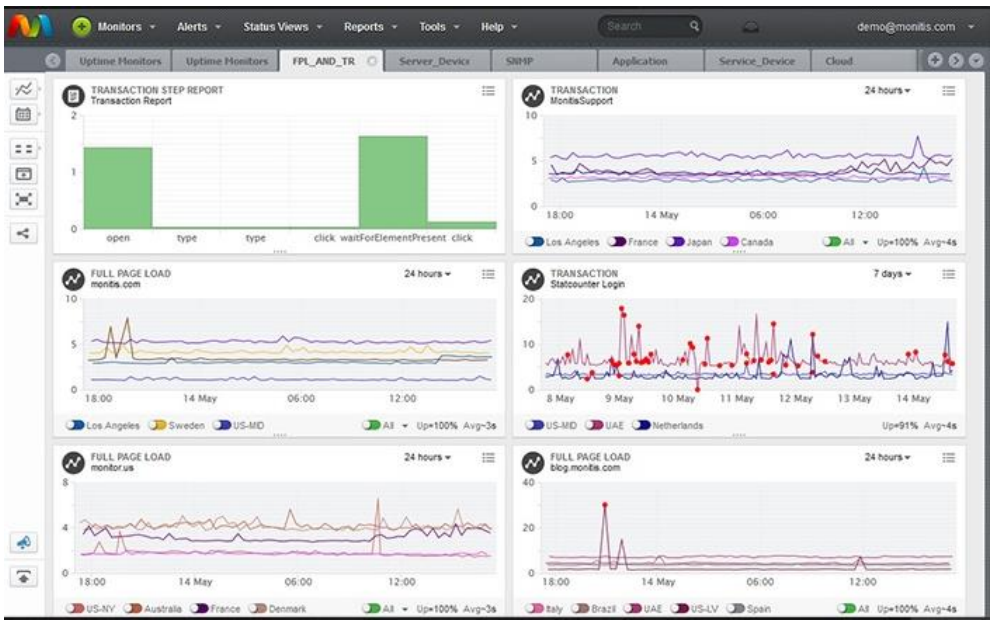


شکل ۴-۲: داشبورد Datadog با نمایش یکپارچه داده‌های شبکه و برنامه‌های ابری.

این ابزار با داشبوردهای پیشرفته، نظارت بر شبکه، سرورها و برنامه‌های ابری را امکان‌پذیر کرده و با بیش از ۴۰۰ ابزار دیگر یکپارچه می‌شود .

۵-۹-۲- Monitis :

Monitisیک ابزار تخصصی برای مانیتورینگ وب‌سایت است که امکان نظارت بر عملکرد سرورها، شبکه و برنامه‌های وب را ارائه می‌دهد. این ابزار، داشبوردی جامع برای ارائه اطلاعات بلادرنگ در مورد وضعیت وب‌سایت و اجزای مرتبط دارد و برای بهینه‌سازی عملکرد و کاهش زمان خرابی بسیار موثر است.



شکل ۵-۲: داشبورد Monitis با تمرکز بر نظارت بر وب‌سایت و عملکرد سرورها.

این ابزار مانیتورینگ وب‌سایت با داشبوردهای جامع، نظارت بر عملکرد وب‌سایت، سرورها و شبکه را در یک پنل واحد ارائه می‌دهد .

فصل سوم

نصب و راه اندازی ابزار ها بر روی Ubuntu و Widows

۱-۳-نصب و راه اندازی ابزار ها بر روی Ubuntu

۱-۱-۳- نصب و راه اندازی Prometheus

برای نصب ابزار Prometheus از دستور wget و لینک دانلود آخرین نسخه از سایت اصلی اقدام می نماییم.

(لینک دانلود آخرین نسخه پرومتئوس:

–prometheus/۲.۵۳.۱https://github.com/prometheus/prometheus/releases/download/v

(tar.gz.۶۴linux-amd.۲.۵۳.۱

```
amir@amir:~$ wget https://github.com/prometheus/prometheus/releases/download/v2.53.1/prometheus-2.53.1.l
inux-amd64.tar.gz
--2024-07-12 17:44:20--  https://github.com/prometheus/prometheus/releases/download/v2.53.1/prometheus-2
.53.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/6838921/a3bb21a2-
dbd7-4ab5-b939-884e332bd7bf?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F2
0240712%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240712T141421Z&X-Amz-Expires=300&X-Amz-Signature=66
7f91ab5332115aa1e9ca5a8d5710a037649c1473e93bafbb0f6944c4ffb7a1&X-Amz-SignedHeaders=host&actor_id=0&key_i
d=0&repo_id=6838921&response-content-disposition=attachment%3B%20filename%3Dprometheus-2.53.1.linux-amd6
4.tar.gz&response-content-type=application%2Foctet-stream [following]
--2024-07-12 17:44:21--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/68
38921/a3bb21a2-dbd7-4ab5-b939-884e332bd7bf?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseasse
tproduction%2F20240712%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240712T141421Z&X-Amz-Expires=300&X-A
mz-Signature=667f91ab5332115aa1e9ca5a8d5710a037649c1473e93bafbb0f6944c4ffb7a1&X-Amz-SignedHeaders=host&a
ctor_id=0&key_id=0&repo_id=6838921&response-content-disposition=attachment%3B%20filename%3Dprometheus-2.
53.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.108.
133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.111.133|:443... conn
ected.
HTTP request sent, awaiting response... 200 OK
Length: 104191695 (99M) [application/octet-stream]
Saving to: 'prometheus-2.53.1.linux-amd64.tar.gz'

prometheus-2.53.1.l 100%[===================>]  99.36M  1.56MB/s    in 68s

2024-07-12 17:45:31 (1.46 MB/s) - 'prometheus-2.53.1.linux-amd64.tar.gz' saved [104191695/104191695]
```

شکل ۱-۳: دانلود Prpmetheus.

نمایش پیغام حاصله به معنی دانلود کامل و صحیح فایل Prometheus می باشد.

حال با استفاده از دستور زیر فایل دانلود شده را از حالت فشرده خارج می کنیم.

```
amir@amir:~$ tar -xvzf prometheus-2.53.1.linux-amd64.tar.gz
prometheus-2.53.1.linux-amd64/
prometheus-2.53.1.linux-amd64/prometheus.yml
prometheus-2.53.1.linux-amd64/prometheus
prometheus-2.53.1.linux-amd64/consoles/
prometheus-2.53.1.linux-amd64/consoles/node-disk.html
prometheus-2.53.1.linux-amd64/consoles/node-overview.html
prometheus-2.53.1.linux-amd64/consoles/node-cpu.html
prometheus-2.53.1.linux-amd64/consoles/node.html
prometheus-2.53.1.linux-amd64/consoles/prometheus-overview.html
prometheus-2.53.1.linux-amd64/consoles/index.html.example
prometheus-2.53.1.linux-amd64/consoles/prometheus.html
prometheus-2.53.1.linux-amd64/LICENSE
prometheus-2.53.1.linux-amd64/promtool
prometheus-2.53.1.linux-amd64/console_libraries/
prometheus-2.53.1.linux-amd64/console_libraries/menu.lib
prometheus-2.53.1.linux-amd64/console_libraries/prom.lib
prometheus-2.53.1.linux-amd64/NOTICE
```

شکل ۲-۳:خروج Prometheus از حالت فشره (Unzip کردن).

حال با دستور cd prometheus-۲.۵۳.۱.linux-amd۶۴ به دایرکتوری مناسب رفته و پرومتئوس را با استفاده از دستور ./prometheus --config.file=prometheus.yml اجرا می کنیم:

نمایش پیغام زیر به این معنی است که پرومتئوس به طور موفقیت آمیز بر روی سیستم شروع کار کرده است.

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ ./prometheus --config.file=prometheus.yml
ts=2024-07-12T14:32:56.836Z caller=main.go:589 level=info msg="No time or size retention was set so using
the default time retention" duration=15d
ts=2024-07-12T14:32:56.836Z caller=main.go:633 level=info msg="Starting Prometheus Server" mode=server ver
sion="(version=2.53.1, branch=HEAD, revision=14cfec3f6048b735e08c1e9c64c8d4211d32bab4)"
ts=2024-07-12T14:32:56.836Z caller=main.go:638 level=info build_context="(go=go1.22.5, platform=linux/amd6
4, user=root@9f8e5b6970da, date=20240710-10:16:27, tags=netgo,builtinassets,stringlabels)"
ts=2024-07-12T14:32:56.836Z caller=main.go:639 level=info host_details="(Linux 6.5.0-41-generic #41-Ubuntu
 SMP PREEMPT_DYNAMIC Mon May 20 15:55:15 UTC 2024 x86_64 amir (none))"
ts=2024-07-12T14:32:56.836Z caller=main.go:640 level=info fd_limits="(soft=1048576, hard=1048576)"
ts=2024-07-12T14:32:56.836Z caller=main.go:641 level=info vm_limits="(soft=unlimited, hard=unlimited)"
ts=2024-07-12T14:32:56.842Z caller=web.go:568 level=info component=web msg="Start listening for connection
s" address=0.0.0.0:9090
ts=2024-07-12T14:32:56.843Z caller=main.go:1148 level=info msg="Starting TSDB ..."
ts=2024-07-12T14:32:56.857Z caller=tls_config.go:313 level=info component=web msg="Listening on" address=[
::]:9090
ts=2024-07-12T14:32:56.857Z caller=tls_config.go:316 level=info component=web msg="TLS is disabled." http2
=false address=[::]:9090
ts=2024-07-12T14:32:56.862Z caller=head.go:626 level=info component=tsdb msg="Replaying on-disk memory map
pable chunks if any"
ts=2024-07-12T14:32:56.862Z caller=head.go:713 level=info component=tsdb msg="On-disk memory mappable chun
ks replay completed" duration=3.135µs
ts=2024-07-12T14:32:56.863Z caller=head.go:721 level=info component=tsdb msg="Replaying WAL, this may take
 a while"
ts=2024-07-12T14:32:56.863Z caller=head.go:793 level=info component=tsdb msg="WAL segment loaded" segment=
0 maxSegment=0
ts=2024-07-12T14:32:56.863Z caller=head.go:830 level=info component=tsdb msg="WAL replay completed" checkp
oint_replay_duration=137.418µs wal_replay_duration=361.559µs wbl_replay_duration=213ns chunk_snapshot_load
_duration=0s mmap_chunk_replay_duration=3.135µs total_replay_duration=693.927µs
ts=2024-07-12T14:32:56.868Z caller=main.go:1169 level=info fs_type=EXT4_SUPER_MAGIC
ts=2024-07-12T14:32:56.876Z caller=main.go:1172 level=info msg="TSDB started"
ts=2024-07-12T14:32:56.876Z caller=main.go:1354 level=info msg="Loading configuration file" filename=prome
theus.yml
ts=2024-07-12T14:32:56.877Z caller=main.go:1391 level=info msg="updated GOGC" old=100 new=75
ts=2024-07-12T14:32:56.877Z caller=main.go:1402 level=info msg="Completed loading of configuration file" f
ilename=prometheus.yml totalDuration=1.354904ms db_storage=2.929µs remote_storage=2.802µs web_handler=733n
s query_engine=2.783µs scrape=509.702µs scrape_sd=37.256µs notify=92.903µs notify_sd=17.315µs rules=2.05µs
 tracing=10.304µs
ts=2024-07-12T14:32:56.877Z caller=main.go:1133 level=info msg="Server is ready to receive web requests."
ts=2024-07-12T14:32:56.877Z caller=manager.go:164 level=info component="rule manager" msg="Starting rule m
anager..."
```

شکل ۳-۳:صحت اجرای Prometheus.

حال باید پرومتئوس را پیکربندی کنیم

بدین منظور با استفاده از دستور sudo nano /etc/systemd/system/prometheus.service فایل prometheus.yml را باز می کنیم و سپس پیکربندی زیر را در آن وارد می نماییم:

```
global:
  scrape_interval: ۱۵s

scrape_configs:
  - job_name: 'prometheus'
    static_configs:
      - targets: ['localhost:۹۰۹۰']
```

```
  GNU nano 7.2                    /etc/systemd/system/prometheus.service
global:
  scrape_interval: 15s

scrape_configs:
  - job_name: 'prometheus'
    static_configs:
      - targets: ['localhost:9090']
```

شکل ۴-۳: پیکربندی Prometheus.

برای پیکربندی کامل تر می توانید دستورات زیر را وارد کنید:

```
[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
    --config.file /etc/prometheus/prometheus.yml \
    --storage.tsdb.path /var/lib/prometheus/ \
    --web.console.templates=/usr/share/prometheus/consoles \
    --web.console.libraries=/usr/share/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

شکل ۵-۳: پیکربندی کامل تر برای Prometheus.

حال با استفاده از دستورات زیر پرومتئوس را راه اندازی و اجرا می کنیم:

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl daemon-reload
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl start prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl enable prometheus
Created symlink /etc/systemd/system/multi-user.target.wants/prometheus.service → /etc/systemd/system/prome
theus.service.
```

شکل ۶-۳: راه اندازی و اجرای Prometheus.

۲-۱-۳- نصب و راه اندازی Node Exporter

برای جمع‌آوری داده‌ها از منابع مختلف، نیاز به نصب و پیکربندی اکسپورترهای مربوطه داریم. به عنوان مثال برای مانیتورینگ شبکه، سیستم‌عامل و وب‌سایت، می‌توان از Node_exporter و Blackbox_exporter استفاده کرد.

Node Exporter برای جمع‌آوری اطلاعات سیستم‌عامل و سخت‌افزار استفاده می‌شود:

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ curl -LO https://github.com/prometheus/node_exporter/releases/d
ownload/v1.1.2/node_exporter-1.1.2.linux-amd64.tar.gz
Command 'curl' not found, but can be installed with:
sudo snap install curl  # version 8.1.2, or
sudo apt  install curl  # version 8.2.1-1ubuntu3.1
See 'snap info curl' for additional versions.
```

شکل ۷-۳: دانلود Node Exporter.

حال در اینجا از دستور Sudo apt update استفاده می کنیم.

این دستور باعث به‌روزرسانی فهرست بسته‌های موجود در سیستم شما می‌شود. این فهرست‌ها شامل اطلاعات درباره بسته‌های موجود در مخازن نرم‌افزاری مورد استفاده شما می‌باشد، از جمله نام بسته، نسخه، وضعیت وجود یا به‌روزرسانی و... است.

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo apt update
Get:1 http://security.ubuntu.com/ubuntu mantic-security InRelease [127 kB]
Hit:2 http://ir.archive.ubuntu.com/ubuntu mantic InRelease
Get:3 http://ir.archive.ubuntu.com/ubuntu mantic-updates InRelease [127 kB]
Get:4 http://ir.archive.ubuntu.com/ubuntu mantic-backports InRelease [127 kB]
Get:5 http://security.ubuntu.com/ubuntu mantic-security/main amd64 Packages [436 kB]
Get:6 http://ir.archive.ubuntu.com/ubuntu mantic-updates/main amd64 Packages [374 kB]
Get:7 http://security.ubuntu.com/ubuntu mantic-security/main Translation-en [110 kB]
Get:8 http://ir.archive.ubuntu.com/ubuntu mantic-updates/main Translation-en [109 kB]
Get:9 http://security.ubuntu.com/ubuntu mantic-security/main amd64 c-n-f Metadata [6,916 B]
Get:10 http://security.ubuntu.com/ubuntu mantic-security/restricted amd64 Packages [405 kB]
Get:11 http://ir.archive.ubuntu.com/ubuntu mantic-updates/main amd64 c-n-f Metadata [9,496 B]
Get:12 http://ir.archive.ubuntu.com/ubuntu mantic-updates/restricted amd64 Packages [235 kB]
Get:13 http://ir.archive.ubuntu.com/ubuntu mantic-updates/restricted Translation-en [35.7 kB]
Get:14 http://ir.archive.ubuntu.com/ubuntu mantic-updates/restricted amd64 c-n-f Metadata [520 B]
Get:15 http://ir.archive.ubuntu.com/ubuntu mantic-updates/universe amd64 Packages [415 kB]
Get:16 http://security.ubuntu.com/ubuntu mantic-security/restricted Translation-en [73.5 kB]
Get:17 http://security.ubuntu.com/ubuntu mantic-security/restricted amd64 c-n-f Metadata [496 B]
Get:18 http://security.ubuntu.com/ubuntu mantic-security/universe amd64 Packages [321 kB]
Get:19 http://ir.archive.ubuntu.com/ubuntu mantic-updates/universe Translation-en [158 kB]
Get:20 http://ir.archive.ubuntu.com/ubuntu mantic-updates/universe amd64 c-n-f Metadata [14.7 kB]
Get:21 http://ir.archive.ubuntu.com/ubuntu mantic-updates/multiverse amd64 Packages [8,852 B]
Get:22 http://ir.archive.ubuntu.com/ubuntu mantic-updates/multiverse Translation-en [2,900 B]
Get:23 http://ir.archive.ubuntu.com/ubuntu mantic-updates/multiverse amd64 c-n-f Metadata [424 B]
Get:24 http://ir.archive.ubuntu.com/ubuntu mantic-backports/universe amd64 c-n-f Metadata [172 B]
Get:25 http://security.ubuntu.com/ubuntu mantic-security/universe Translation-en [129 kB]
Get:26 http://security.ubuntu.com/ubuntu mantic-security/universe amd64 c-n-f Metadata [11.6 kB]
Get:27 http://security.ubuntu.com/ubuntu mantic-security/multiverse amd64 c-n-f Metadata [244 B]
Fetched 3,238 kB in 3s (933 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
83 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

شکل ۸-۳: به‌روزرسانی لیست بسته‌های نرم‌افزاری در سیستم(sudo apt update).

۳-۱-۳- نصب و راه اندازی Curl:

Curl یک ابزار خط فرمان است که برای ارسال و دریافت داده‌ها از طریق پروتکل‌های شبکه مانند HTTP، HTTPS، FTP و ... استفاده می‌شود. این ابزار بسیار مفید است برای دسترسی به منابع اینترنتی مانند دانلود فایل‌ها، ارسال درخواست‌های API، دریافت اطلاعات از وبسایت‌ها و غیره. در پروژه‌های بسیاری، مانند نصب نرم‌افزارها، مدیریت کانفیگ‌ها، اجرای اسکریپت‌های آنلاین، ارسال و دریافت اطلاعات از سرورها، curl استفاده می‌شود.

در پروژه‌ای که شما از آن سوال کرده‌اید، Curl به دلیل قابلیت‌هایش برای دانلود فایل‌ها (مانند دانلود Node Exporter) و ارسال درخواست‌های HTTP برای اینترکانکت با سرویس‌های خارجی (مانند Grafana) نصب شده است.

به طور خلاصه، Curl ابزاری است که به شما امکان می‌دهد تا با استفاده از خط فرمان، به سادگی با منابع اینترنتی ارتباط برقرار کنید و اطلاعات را دریافت یا ارسال کنید.

حال با استفاده از وارد کردن دستور زیر در کامند لاین Curl را نصب می کنیم:

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo apt install curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 83 not upgraded.
Need to get 216 kB of archives.
After this operation, 490 kB of additional disk space will be used.
Get:1 http://ir.archive.ubuntu.com/ubuntu mantic-updates/main amd64 curl amd64 8.2.1-1ubuntu3.3 [216 kB]
Fetched 216 kB in 0s (485 kB/s)
Selecting previously unselected package curl.
(Reading database ... 194217 files and directories currently installed.)
Preparing to unpack .../curl_8.2.1-1ubuntu3.3_amd64.deb ...
Unpacking curl (8.2.1-1ubuntu3.3) ...
Setting up curl (8.2.1-1ubuntu3.3) ...
Processing triggers for man-db (2.11.2-3) ...
```

شکل ۹-۳: نصب Curl در ترمینال لینوکس.

سپس با استفاده از دستور زیر Node Exporter را دانلود می کنیم.

از Node Exporter برای جمع‌آوری اطلاعات سیستم‌عامل و سخت‌افزار استفاده می‌شود:

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ curl -LO https://github.com/prometheus/node_exporter/releases/
ownload/v1.1.2/node_exporter-1.1.2.linux-amd64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 9029k  100 9029k    0     0  1090k      0  0:00:08  0:00:08 --:--:-- 1658k
```

شکل ۱۰-۳: نصب Node Exporter در ترمینال لینوکس.

این دستور فایل فشرده node_exporter-۱.۱.۲.linux-amd۶۴.tar.gz. را استخراج می کند و محتوای آن را در دایرکتوری فعلی قرار می‌دهد.

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ tar -xvzf node_exporter-1.1.2.linux-amd64.tar.gz
node_exporter-1.1.2.linux-amd64/
node_exporter-1.1.2.linux-amd64/LICENSE
node_exporter-1.1.2.linux-amd64/NOTICE
node_exporter-1.1.2.linux-amd64/node_exporter
```

شکل ۱۱-۳: استخراج Node Exporter در ترمینال لینوکس.

این دستور به دایرکتوری استخراج‌شده node_exporter-۱.۱.۲.linux-amd۶۴. تغییر مسیر می‌دهد.

دستورات ابتدا فایل فشرده node_exporter-۱.۱.۲.linux-amd۶۴.tar.gz. را استخراج کرده و سپس به دایرکتوری استخراج‌شده می‌روند.

حال با استفاده از دستورات زیر دستورات ابتدا فایل باینری node_exporter را به مکان مناسب منتقل می‌کنند و سپس فایلی برای تنظیمات سرویس node_exporter ایجاد می‌کنند تا بتوان آن را به عنوان یک سرویس systemd مدیریت کرد.

```
amir@amir:~$ cd node_exporter-1.1.2.linux-amd64
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo mv node_exporter /usr/local/bin/
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo nano /etc/systemd/system/node_exporter.service
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo nano /etc/systemd/system/node_exporter.service
```

شکل ۱۲-۳: انتقال به دایرکتوری مناسب برای مدیریت.

این اطلاعات مربوط به فایل پیکربندی سرویس systemd برای node_exporter است. این فایل به سیستم‌عامل لینوکس کمک می‌کند تا سرویس node_exporter را مدیریت کند.

[Unit] این بخش شامل توضیحات و وابستگی‌های سرویس است.

- Description: توضیح مختصری از سرویس.
- Wants و After: این خطوط وابستگی‌ها را مشخص می‌کنند.

[Service] این بخش شامل تنظیمات مربوط به اجرای سرویس است.

- User و Group: این تنظیمات مشخص می‌کنند که سرویس به عنوان کدام کاربر و گروه اجرا شود.
- Type: نوع سرویس را مشخص می‌کند.
- ExecStart: فرمانی که برای شروع سرویس اجرا می‌شود.

[Install] این بخش شامل تنظیمات مربوط به نصب و راه‌اندازی سرویس است.

- WantedBy: مشخص می‌کند که این سرویس در چه سطح run-level فعال شود.

```
  GNU nano 7.2                    /etc/systemd/system/node_exporter.service
[Unit]
Description=Node Exporter
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/node_exporter

[Install]
WantedBy=multi-user.target
```

شکل ۱۳-۳: پیکر بندی Node Exporter .

سپس دستورات زیر را اجرا می کنیم تا سرویس Node Exporter راه‌اندازی شود:

```
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo systemctl daemon-reload
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo systemctl start node_exporter
amir@amir:~/node_exporter-1.1.2.linux-amd64$ sudo systemctl enable node_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/node_exporter.service → /etc/systemd/system/node_ex
porter.service.
```

شکل ۱۴-۳: راه اندازی Node Exporter.

٤-۱-۳- نصب و راه اندازی Blackbox Exporter :

Blackbox Exporter یکی از ابزارهای Prometheus است که برای مانیتورینگ وب‌سایت‌ها و سایر سرویس‌های اینترنتی استفاده می‌شود. این ابزار به شما امکان می‌دهد تا سرویس‌های شبکه را با استفاده از پروتکل‌های مختلف (مانند HTTP، HTTPS، DNS، TCP و ICMP) بررسی و مانیتور کنید.

با پیکربندی مناسب و ادغام آن با Prometheus، می‌توانید نظارت کامل‌تری بر روی سرویس‌های خود داشته باشید و مشکلات احتمالی را سریع‌تر شناسایی و رفع کنید.

```
amir@amir:~/node_exporter-1.1.2.linux-amd64$ curl -LO https://github.com/prometheus/blackbox_exporter/releases/
download/v0.19.0/blackbox_exporter-0.19.0.linux-amd64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 8673k  100 8673k    0     0   862k      0  0:00:10  0:00:10 --:--:-- 1122k
```

شکل ۱۵-۳: پیکربندی و انتقال Blackbox Exporter به دایرکتوری مناسب.

این دستورات ابتدا فایل فشرده blackbox_exporter-۰.۱۹.۰.linux-amd۶۴.tar.gz. را استخراج می‌کنند و سپس به دایرکتوری استخراج‌شده blackbox_exporter-۰.۱۹.۰.linux-amd۶۴ می‌روند.

```
amir@amir:~/node_exporter-1.1.2.linux-amd64$ tar xvf blackbox_exporter-0.19.0.linux-amd64.tar.gz
blackbox_exporter-0.19.0.linux-amd64/
blackbox_exporter-0.19.0.linux-amd64/blackbox.yml
blackbox_exporter-0.19.0.linux-amd64/LICENSE
blackbox_exporter-0.19.0.linux-amd64/NOTICE
blackbox_exporter-0.19.0.linux-amd64/blackbox_exporter
```

شکل ۱۶-۳: تعیین دسترسی Blackbox Exporter.

در این دایرکتوری، شما به فایل‌ها و باینری‌های blackbox_exporter دسترسی خواهید داشت و می‌توانید مراحل بعدی نصب و پیکربندی را انجام دهید:

```
amir@amir:~/node_exporter-1.1.2.linux-amd64$ tar xvf blackbox_exporter-0.19.0.linux-amd64.tar.gz
blackbox_exporter-0.19.0.linux-amd64/
blackbox_exporter-0.19.0.linux-amd64/blackbox.yml
blackbox_exporter-0.19.0.linux-amd64/LICENSE
blackbox_exporter-0.19.0.linux-amd64/NOTICE
blackbox_exporter-0.19.0.linux-amd64/blackbox_exporter
amir@amir:~/node_exporter-1.1.2.linux-amd64$ cd blackbox_exporter-0.19.0.linux-amd64
amir@amir:~/node_exporter-1.1.2.linux-amd64/blackbox_exporter-0.19.0.linux-amd64$ sudo mv blackbox_exporter /us
r/local/bin/
```

شکل ۱۷-۳: تعیین دایرکتوری Blackbox Exporter.

این دستور فایل باینری blackbox_exporter را به دایرکتوری /usr/local/bin/ منتقل می‌کند، که معمولاً برای باینری‌های کاربران محلی استفاده می‌شود.

```
amir@amir:~/node_exporter-1.1.2.linux-amd64/blackbox_exporter-0.19.0.linux-amd64$ sudo systemctl daemon-reload
amir@amir:~/node_exporter-1.1.2.linux-amd64/blackbox_exporter-0.19.0.linux-amd64$ sudo systemctl start blackbox
_exporter
amir@amir:~/node_exporter-1.1.2.linux-amd64/blackbox_exporter-0.19.0.linux-amd64$ sudo systemctl enable blackbo
x_exporter
Created symlink /etc/systemd/system/multi-user.target.wants/blackbox_exporter.service → /etc/systemd/system/bla
ckbox_exporter.service.
```

شکل ۱۸-۳: مدیریت و اجرای Blackbox Exporter.

این دستور فایل blackbox_exporter.service را با ویرایشگر متنی nano باز می‌کند. این فایل شامل تنظیمات سرویس systemd برای blackbox_exporter خواهد بود. این سرویس به سیستم اجازه می‌دهد تا به راحتی blackbox_exporter را مدیریت کند (مانند شروع، توقف، و راه‌اندازی مجدد).

```
amir@amir:~$ nano /path/to/prometheus.yml
amir@amir:~$ sudo find / -name prometheus.yml
find: '/run/user/1000/gvfs': Permission denied
find: '/run/user/1000/doc': Permission denied
/home/amir/prometheus-2.53.1.linux-amd64/prometheus.yml
amir@amir:~$ ^C
amir@amir:~$ sudo nano /home/amir/prometheus-2.53.1.linux-amd64/prometheus.yml
amir@amir:~$ sudo nano /home/amir/prometheus-2.53.1.linux-amd64/prometheus.yml
amir@amir:~$ sudo systemctl restart prometheus
```

شکل ۱۹-۳: اجرای مجدد برای همگام سازی سرویس ها.

در فایل blackbox_exporter.service، اطلاعات زیر را وارد می کنیم:

[Unit] این بخش شامل توضیحات و وابستگی‌های سرویس است:

- Description: توضیح مختصری از سرویس.
- Wants و After: این خطوط وابستگی‌های سرویس را مشخص می‌کنند.

[Service] این بخش شامل تنظیمات مربوط به اجرای سرویس است:

- User و Group: این تنظیمات مشخص می‌کنند که سرویس به عنوان کدام کاربر و گروه اجرا شود.
- Type: نوع سرویس را مشخص می‌کند.
- ExecStart: فرمانی که برای شروع سرویس اجرا می‌شود.

[Install] این بخش شامل تنظیمات مربوط به نصب و راه‌اندازی سرویس است:

- WantedBy: مشخص می‌کند که این سرویس در چه سطح run-level فعال شود.

```
[Unit]
Description=Blackbox Exporter
Wants=network-online.target
After=network-online.target

[Service]
```

```
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/blackbox_exporter \
    --config.file /etc/prometheus/blackbox.yml

[Install]
WantedBy=multi-user.target
```

در فایل پیکربندی prometheus.yml، تنظیمات مربوط به اکسپورترها را اضافه سپس Prometheus را (با استفاده از دستور sudo systemctl restart Prometheus) ری‌استارت می کنیم:

```
  GNU nano 7.2                                  /path/to/prometheus.yml *
global:
  scrape_interval: 15s

scrape_configs:
  - job_name: 'node_exporter'
    static_configs:
      - targets: ['localhost:9100']

  - job_name: 'blackbox_exporter'
    metrics_path: /probe
    params:
      module: [http_2xx]
    static_configs:
      - targets:
        - http://example.com
    relabel_configs:
      - source_labels: [address]
        target_label: param_target
      - source_labels: [__param_target]
        target_label: instance
      - target_label: __address
        replacement: localhost:9115
```

شکل ۲۰-۳: اضافه کردن تنظیمات اکسپوتر ها به پیکربندی prometheus.yml.

۵-۱-۳- نصب و راه اندازی Grafana :

حال با استفاده از دستور wget برای دریافت کلید عمومی و اضافه کردن آن به سیستم استفاده می کنیم:

این دستور کلید عمومی مخزن بسته‌های نرم‌افزاری Grafana را دریافت و به سیستم شما اضافه می‌کند. این کلید عمومی برای تایید صحت و یکپارچگی بسته‌های نرم‌افزاری استفاده می‌شود. با این کلید، سیستم شما می‌تواند مطمئن شود که بسته‌های نرم‌افزاری از یک منبع معتبر و بدون تغییر دریافت شده‌اند.

```
amir@amir:~$ wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see ap
t-key(8)).
OK
```

شکل ۲۱-۳: دریافت Grafana .

با اجرای این دستور، کلید عمومی GPG مربوط به مخزن بسته‌های نرم‌افزاری Grafana به سیستم شما اضافه می‌شود. این کلید به سیستم شما اجازه می‌دهد تا بسته‌های نرم‌افزاری دانلود شده از مخزن Grafana را تایید کرده و مطمئن شود که این بسته‌ها از منبع معتبر و بدون تغییر دریافت شده‌اند.

حال در اینجا مجددا از دستور sudo apt update استفاده می کنیم.

این دستور باعث به‌روزرسانی فهرست بسته‌های موجود در سیستم شما می‌شود. این فهرست‌ها شامل اطلاعات درباره بسته‌های موجود در مخازن نرم‌افزاری مورد استفاده شما می‌باشد، از جمله نام بسته، نسخه، وضعیت وجود یا به‌روزرسانی و... است.

```
amir@amir:~$ sudo apt update
Hit:1 http://ir.archive.ubuntu.com/ubuntu mantic InRelease
Hit:2 http://ir.archive.ubuntu.com/ubuntu mantic-updates InRelease
Hit:3 http://ir.archive.ubuntu.com/ubuntu mantic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu mantic-security InRelease
Ign:5 https://packages.grafana.com/oss/deb stable InRelease
Ign:5 https://packages.grafana.com/oss/deb stable InRelease
Get:5 https://packages.grafana.com/oss/deb stable InRelease [7,661 B]
Get:6 https://packages.grafana.com/oss/deb stable/main amd64 Packages [261 kB]
Fetched 261 kB in 7s (39.2 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
83 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: https://packages.grafana.com/oss/deb/dists/stable/InRelease: Key is stored in lega
cy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key
(8) for details.
```

شکل ۲۲-۳: به‌روزرسانی لیست بسته‌های نرم‌افزاری در سیستم(sudo apt update).

حال با دستور زیر Grafana را نصب می کنیم:

```
amir@amir:~$ sudo apt install grafana
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  musl
The following NEW packages will be installed:
  grafana musl
0 upgraded, 2 newly installed, 0 to remove and 83 not upgraded.
Need to get 118 MB of archives.
After this operation, 438 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ir.archive.ubuntu.com/ubuntu mantic/universe amd64 musl amd64 1.2.3-1 [4
04 kB]
Ign:2 https://packages.grafana.com/oss/deb stable/main amd64 grafana amd64 11.1.0
Get:2 https://packages.grafana.com/oss/deb stable/main amd64 grafana amd64 11.1.0 [11
7 MB]
Fetched 118 MB in 8min 26s (232 kB/s)
Selecting previously unselected package musl:amd64.
(Reading database ... 194223 files and directories currently installed.)
Preparing to unpack .../musl_1.2.3-1_amd64.deb ...
Unpacking musl:amd64 (1.2.3-1) ...
Selecting previously unselected package grafana.
Preparing to unpack .../grafana_11.1.0_amd64.deb ...
Unpacking grafana (11.1.0) ...
Setting up musl:amd64 (1.2.3-1) ...
Setting up grafana (11.1.0) ...
info: Selecting UID from range 100 to 999 ...

info: Adding system user `grafana' (UID 126) ...
info: Adding new user `grafana' (UID 126) with group `grafana' ...
info: Not creating home directory `/usr/share/grafana'.
### NOT starting on installation, please execute the following statements to configur
e grafana to start automatically using systemd
 sudo /bin/systemctl daemon-reload
 sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
 sudo /bin/systemctl start grafana-server
Processing triggers for man-db (2.11.2-3) ...
```

شکل ۲۳-۳: نصب Grafana.

گرافانا یک ابزار متن باز (open-source) برای مانیتورینگ و داشبوردسازی داده‌ها و سیستم‌ها است که به ویژه برای مانیتورینگ سیستم‌های زیرساخت فناوری اطلاعات (IT infrastructure) مورد استفاده قرار می‌گیرد. این ابزار ابتدا توسط شرکت Grafana Labs توسعه یافته است و در حال حاضر تحت لایسنس Apache ۲.۰ منتشر

می‌شود و به کمک آن می‌توان به صورت موثر و کارآمد عملکرد سیستم‌ها و شبکه‌های مختلف را نظارت و مدیریت کرد.

```
amir@amir:~$ sudo systemctl start grafana-server
amir@amir:~$ sudo systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/syst
emd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service →
/lib/systemd/system/grafana-server.service.
```

شکل ۲۴-۳: نصب و جرای Grafana بر روی سیستم.

این دستورات به شما کمک می‌کنند تا Grafana را در سیستم خود نصب کرده و اجرا کنید. با اضافه کردن مخزن Grafana، شما قادر به نصب آخرین نسخه Grafana با استفاده از مدیریت بسته‌های APT خواهید بود و با شروع سرویس، می‌توانید به طور مستقیم به واسط کاربری Grafana دسترسی داشته باشید تا داشبوردها و گزارش‌های مورد نظر خود را ایجاد و مدیریت کنید.

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo mv ~/prometheus-2.53.1.linux-amd64/prometheus /usr/
local/bin/prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo chown -R prometheus:prometheus /usr/local/bin/prome
theus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo mkdir -p /etc/prometheus /var/lib/prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo chown -R prometheus:prometheus /etc/prometheus /var
/lib/prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo mv ~/prometheus-2.53.1.linux-amd64/prometheus.yml /
etc/prometheus/
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo chown prometheus:prometheus /etc/prometheus/prometh
eus.yml
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo nano /etc/systemd/system/prometheus.service
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl daemon-reload
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl start prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl enable prometheus
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl status prometheus
```

شکل ۲۵-۳: راه اندازی سرویس Grafana.

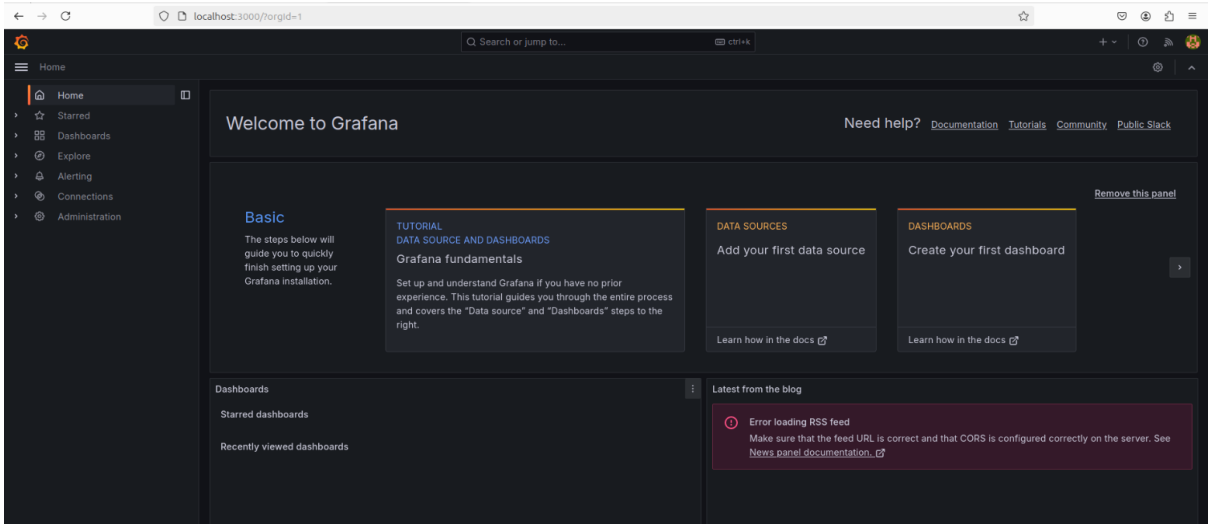
به Grafana وارد شوید (معمولاً در آدرس http://localhost:۳۰۰۰).

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl status grafana-server
[sudo] password for amir:
● grafana-server.service - Grafana instance
     Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; prese>
     Active: active (running) since Fri 2024-07-12 20:15:00 +0330; 47min ago
       Docs: http://docs.grafana.org
   Main PID: 1027 (grafana)
      Tasks: 14 (limit: 4569)
     Memory: 171.2M
        CPU: 4.685s
     CGroup: /system.slice/grafana-server.service
             └─1027 /usr/share/grafana/bin/grafana server --config=/etc/grafana>

Jul 12 20:15:05 amir grafana[1027]: logger=plugin.angulardetectorsprovider.dyna>
Jul 12 20:15:05 amir grafana[1027]: logger=grafana-apiserver t=2024-07-12T20:15>
Jul 12 20:15:05 amir grafana[1027]: logger=grafana-apiserver t=2024-07-12T20:15>
Jul 12 20:15:50 amir grafana[1027]: logger=infra.usagestats t=2024-07-12T20:15:>
Jul 12 20:16:06 amir grafana[1027]: logger=plugin.signature.key_retriever t=202>
Jul 12 20:25:05 amir grafana[1027]: logger=cleanup t=2024-07-12T20:25:05.183103>
Jul 12 20:35:05 amir grafana[1027]: logger=cleanup t=2024-07-12T20:35:05.180569>
Jul 12 20:45:05 amir grafana[1027]: logger=cleanup t=2024-07-12T20:45:05.179775>
Jul 12 20:45:50 amir grafana[1027]: logger=infra.usagestats t=2024-07-12T20:45:>
Jul 12 20:55:05 amir grafana[1027]: logger=cleanup t=2024-07-12T20:55:05.179971>
```

شکل ۲۶-۳: صحت اجرای Grafana.

یک Data Source جدید ایجاد کنید و Prometheus را انتخاب کنید.



شکل ۲۷-۳: محیط اجرای Grafana.

برای بررسی صحت نصب پرومتئوس از دستور زیر استفاده می کنیم:

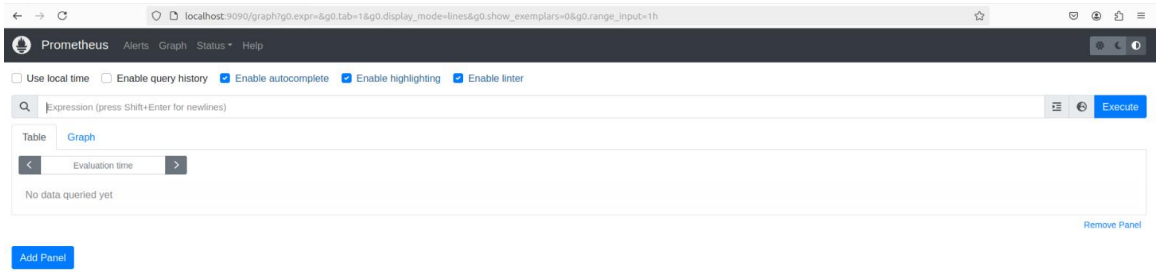
```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl status prometheus
● prometheus.service - Prometheus
     Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; preset: e>
     Active: active (running) since Fri 2024-07-12 21:01:26 +0330; 1min 18s ago
   Main PID: 4264 (prometheus)
      Tasks: 7 (limit: 4569)
     Memory: 20.2M
        CPU: 173ms
     CGroup: /system.slice/prometheus.service
             └─4264 /usr/local/bin/prometheus --config.file /etc/prometheus/pro>

Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.869Z caller=head.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.870Z caller=head.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.870Z caller=head.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.874Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.874Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.875Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.875Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.875Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.875Z caller=main.>
Jul 12 21:01:26 amir prometheus[4264]: ts=2024-07-12T17:31:26.875Z caller=manag>
```

شکل ۲۸-۳: تست اجرای Prometheus.

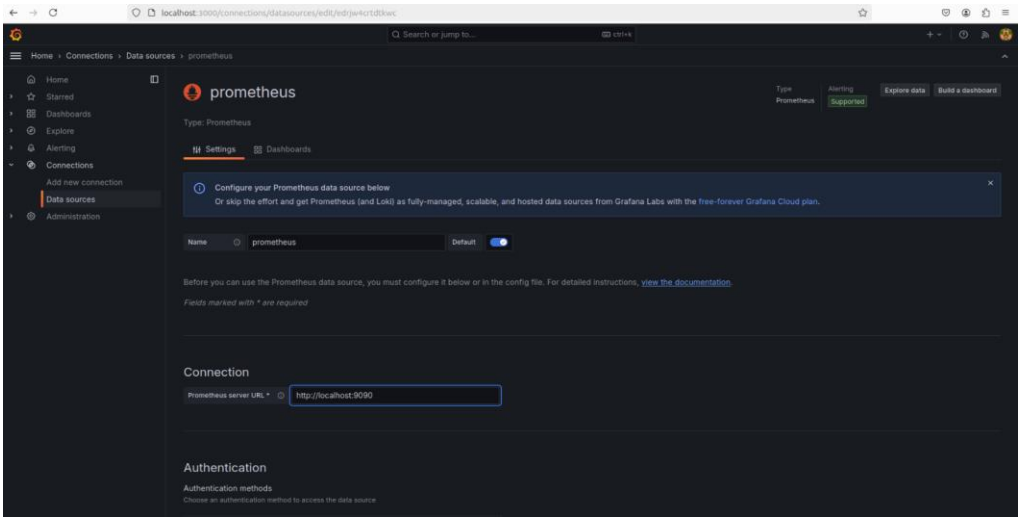
URL سرویس Prometheus را وارد کنید (معمولاً http://localhost:9090).

با این دستور میتوانید از صحت نصب پرومتئوس هم مطمئن شوید:



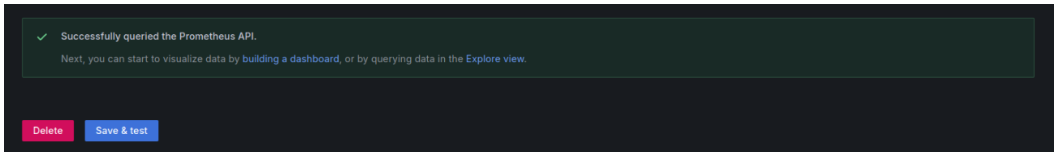
شکل ۲۹-۳: صحت نصب Prometheus.

حال که از صحت نصب گرافانا و پرومتئوس مطمئن شدیم با وارد کردن پورت پرومتئوس داخل گرافانا آنها را بهم متصل می کنیم: (http://localhost9090).



شکل ۳۰-۳: اتصال Prometheus به Grafana.

حال با انتخاب گزینه سیو و تست و دریافت پیغام تایید از صحت اتصال اطمینان حاصل می کنیم.



شکل ۳۱-۳: تست صحت اتصال.

بررسی صحت عملکرد نود اکسپورتر:

```
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl start node_exporter
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl enable node_exporter
amir@amir:~/prometheus-2.53.1.linux-amd64$ sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset: enabled)
     Active: active (running) since Fri 2024-07-12 23:38:50 +0330; 15s ago
   Main PID: 7564 (node_exporter)
      Tasks: 4 (limit: 4569)
     Memory: 13.2M
        CPU: 37ms
     CGroup: /system.slice/node_exporter.service
             └─7564 /usr/local/bin/node_exporter

Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=node_expor>
Jul 12 23:38:50 amir node_exporter[7564]: level=info ts=2024-07-12T20:08:50.240Z caller=tls_config>
```

شکل ۳۲-۳: صحت عملکرد Node Exporter.

پیکر بندی جدید فایل Prometheus.yml :

```
  GNU nano 7.2                    /etc/prometheus/prometheus.yml *
global:
  scrape_interval: 15s # تنظیم فاصله زمانی برای استخراج داده‌ها به هر 15 ثانیه
  evaluation_interval: 15s # ارزیابی قوانین هر 15 ثانیه

# پیکربندی Alertmanager
alerting:
  alertmanagers:
    - static_configs:
        - targets:
          # - alertmanager:9093

# جهانی 'evaluation_interval' بارگیری قوانین و ارزیابی دوره‌ای آن‌ها بر اساس
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"

# پیکربندی استخراج داده‌ها (scrape)
scrape_configs:
  - job_name: "prometheus"
    static_configs:
      - targets: ["localhost:9090"]

  - job_name: "node_exporter"
    static_configs:
      - targets: ["localhost:9100"]
```

شکل ۳۳-۳: پیکربندی نهایی Prometheus.

٦-۱-۳- نصب و راه اندازی Nginx :

```
amir@amir:~$ sudo apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  nginx-common
Suggested packages:
  fcgiwrap nginx-doc
The following NEW packages will be installed:
  nginx nginx-common
0 upgraded, 2 newly installed, 0 to remove and 36 not upgraded.
Need to get 565 kB of archives.
After this operation, 1,604 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ir.archive.ubuntu.com/ubuntu mantic/main amd64 nginx-common all 1.24.0-1ubuntu1 [44.8 kB]
Get:2 http://ir.archive.ubuntu.com/ubuntu mantic/main amd64 nginx amd64 1.24.0-1ubuntu1 [521 kB]
Fetched 565 kB in 1s (985 kB/s)
Preconfiguring packages ...
Selecting previously unselected package nginx-common.
(Reading database ... 203793 files and directories currently installed.)
Preparing to unpack .../nginx-common_1.24.0-1ubuntu1_all.deb ...
Unpacking nginx-common (1.24.0-1ubuntu1) ...
Selecting previously unselected package nginx.
Preparing to unpack .../nginx_1.24.0-1ubuntu1_amd64.deb ...
Unpacking nginx (1.24.0-1ubuntu1) ...
Setting up nginx-common (1.24.0-1ubuntu1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /lib
/systemd/system/nginx.service.
Setting up nginx (1.24.0-1ubuntu1) ...
 * Upgrading binary nginx                                          [ OK ]
Processing triggers for man-db (2.11.2-3) ...
Processing triggers for ufw (0.36.2-1) ...
amir@amir:~$
```

شکل ۳۴-۳: نصب Nginx.

فعال کردن و سنجش صحت:

```
amir@amir:~$ sudo systemctl start nginx
amir@amir:~$ sudo systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nginx
amir@amir:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Sun 2024-07-14 15:14:02 +0330; 2min 5s ago
       Docs: man:nginx(8)
   Main PID: 4298 (nginx)
      Tasks: 5 (limit: 9400)
     Memory: 3.7M
        CPU: 71ms
     CGroup: /system.slice/nginx.service
             ├─4298 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─4301 "nginx: worker process"
             ├─4302 "nginx: worker process"
             ├─4303 "nginx: worker process"
             └─4304 "nginx: worker process"

Jul 14 15:14:02 amir systemd[1]: Starting nginx.service - A high performance web server and a reverse prox>
Jul 14 15:14:02 amir systemd[1]: Started nginx.service - A high performance web server and a reverse proxy>
lines 1-17/17 (END)
amir@amir:~$
```

شکل ۳۵-۳: تست صحت و فعال سازی Nginx.

۷-۱-۳- نصب و راه اندازی InfluxDB :

InfluxDB یک پایگاه داده سری زمانی (Time Series Database یا TSDB) است که به‌طور خاص برای مدیریت و ذخیره‌سازی داده‌های سری زمانی طراحی شده است. داده‌های سری زمانی شامل داده‌هایی هستند که در طول زمان جمع‌آوری می‌شوند، مانند داده‌های سنسورها، داده‌های عملکرد سیستم، داده‌های مالی و غیره. InfluxDB به‌عنوان یک پایگاه داده با کارایی بالا و برای بارگذاری، ذخیره‌سازی و پرس‌وجوی داده‌های سری زمانی بهینه‌سازی شده است.

ابتدا باید مخزن InfluxData را به سیستم خود اضافه کنید:

```
amir@amir:~$ wget -qO- https://repos.influxdata.com/influxdb.key | sudo apt-key
add -
[sudo] password for amir:
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (s
ee apt-key(8)).
OK
amir@amir:~$
```

شکل ۳۶-۳: اضافه کردن مخازن InfluxDB.

سپس مخزن InfluxData را به لیست مخازن سیستم اضافه کنید:

```
amir@amir:~$ source /etc/os-release
amir@amir:~$ echo "deb https://repos.influxdata.com/ubuntu $VERSION_CODENAME sta
ble" | sudo tee /etc/apt/sources.list.d/influxdata.list
deb https://repos.influxdata.com/ubuntu mantic stable
```

شکل ۳۷-۳: اضافه کردن مخازن InfluxDB به مخازن سیستم.

حالا مخازن را بروزرسانی کنید و InfluxDB را نصب کنید:

بروز رسانی:

```
amir@amir:~$ sudo apt-get update
Hit:1 http://security.ubuntu.com/ubuntu mantic-security InRelease
Hit:2 http://ir.archive.ubuntu.com/ubuntu mantic InRelease
Hit:3 http://ir.archive.ubuntu.com/ubuntu mantic-updates InRelease
Hit:4 http://ir.archive.ubuntu.com/ubuntu mantic-backports InRelease
Ign:5 https://packages.grafana.com/oss/deb stable InRelease
Ign:6 https://repos.influxdata.com/ubuntu mantic InRelease
Err:7 https://repos.influxdata.com/ubuntu mantic Release
  404  Not Found [IP: 18.239.83.124 443]
Get:5 https://packages.grafana.com/oss/deb stable InRelease [7,661 B]
Get:8 https://packages.grafana.com/oss/deb stable/main amd64 Packages [262 kB]
Reading package lists... Done
E: The repository 'https://repos.influxdata.com/ubuntu mantic Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
W: https://packages.grafana.com/oss/deb/dists/stable/InRelease: Key is stored in legacy trusted.gpg
keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.
```

شکل ۳۸-۳: به‌روزرسانی لیست بسته‌های نرم‌افزاری در سیستم(sudo apt update).

نصب:

```
amir@amir:~$ sudo apt-get install influxdb
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  influxdb
0 upgraded, 1 newly installed, 0 to remove and 36 not upgraded.
Need to get 5,703 kB of archives.
After this operation, 18.6 MB of additional disk space will be used.
Get:1 http://ir.archive.ubuntu.com/ubuntu mantic-updates/universe amd64 influxdb amd64 1.6.7~rc0-2ub
untu0.1 [5,703 kB]
Fetched 5,703 kB in 5s (1,177 kB/s)
Selecting previously unselected package influxdb.
(Reading database ... 204548 files and directories currently installed.)
Preparing to unpack .../influxdb_1.6.7~rc0-2ubuntu0.1_amd64.deb ...
Unpacking influxdb (1.6.7~rc0-2ubuntu0.1) ...
Setting up influxdb (1.6.7~rc0-2ubuntu0.1) ...
info: Selecting UID from range 100 to 999 ...

info: Adding system user `influxdb' (UID 127) ...
info: Adding new user `influxdb' (UID 127) with group `nogroup' ...
info: Not creating home directory `/var/lib/influxdb'.
info: Selecting GID from range 100 to 999 ...
info: Adding group `influxdb' (GID 132) ...
info: Adding user `influxdb' to group `influxdb' ...
Created symlink /etc/systemd/system/influxd.service → /lib/systemd/system/influxdb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/influxdb.service → /lib/systemd/system/i
nfluxdb.service.
Processing triggers for man-db (2.11.2-3) ...
```

شکل ۳-۳۹: نصب InfluxDB

سرویس InfluxDB را شروع کنید و اطمینان حاصل کنید که سرویس به صورت خودکار با هر بار راه‌اندازی سیستم شروع می‌شود:

```
amir@amir:~$ sudo systemctl start influxdb
amir@amir:~$ sudo systemctl enable influxdb
Synchronizing state of influxdb.service with SysV service script with /lib/systemd/systemd-sysv-inst
all.
Executing: /lib/systemd/systemd-sysv-install enable influxdb
```

شکل ۳-۴۰: فعال سازی خودکار InfluxDB.

می‌توانید با استفاده از دستورات زیر بررسی کنید که InfluxDB به درستی نصب شده و در حال اجرا است:

```
amir@amir:~$ sudo systemctl status influxdb
● influxdb.service - InfluxDB is an open-source, distributed, time series database
     Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; preset: enabled)
     Active: active (running) since Mon 2024-07-15 16:27:20 +0330; 2min 27s ago
       Docs: man:influxd(1)
   Main PID: 9635 (influxd)
      Tasks: 9 (limit: 9400)
     Memory: 6.7M
        CPU: 107ms
     CGroup: /system.slice/influxdb.service
             └─9635 /usr/bin/influxd -config /etc/influxdb/influxdb.conf

Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745512Z lvl=info msg="Registered diagnos>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745538Z lvl=info msg="Starting precreati>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745553Z lvl=info msg="Starting snapshot >
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745561Z lvl=info msg="Starting continuou>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745573Z lvl=info msg="Starting HTTP serv>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745579Z lvl=info msg="opened HTTP access>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745667Z lvl=info msg="Storing statistics>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745698Z lvl=info msg="Listening on HTTP">
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745722Z lvl=info msg="Starting retention>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.746150Z lvl=info msg="Listening for sign>
```

شکل ۳-۴۱: تست صحت و اجرای InfluxDB.

فایل پیکربندی InfluxDB معمولاً در مسیر /etc/influxdb/influxdb.conf قرار دارد. شما می‌توانید این فایل را باز کرده و تنظیمات مورد نیاز خود را اعمال کنید.

```
amir@amir:~$ sudo nano /etc/influxdb/influxdb.conf
```

شکل ۴۲-۳: پیکر بندی InfluxDB.

[meta]

"dir = "/var/lib/influxdb/meta

[data]

"dir = "/var/lib/influxdb/data

"wal-dir = "/var/lib/influxdb/wal

[http]

enabled = true

"۸۰۸۶bind-address = ":

auth-enabled = false

۸-۱-۳- نصب و راه اندازی Loki :

Loki یک سیستم لاگینگ توزیع‌شده، مقیاس‌پذیر و قابل اعتماد است که توسط Grafana Labs توسعه داده شده است. Loki به‌طور خاص برای شاخص‌گذاری و جستجوی لاگ‌ها طراحی شده و تمرکز اصلی آن بر روی کارایی و مقیاس‌پذیری بالا با استفاده از حداقل منابع است. برخلاف سیستم‌های لاگینگ سنتی که داده‌های لاگ را به‌شدت شاخص‌گذاری می‌کنند، Loki تنها برچسب‌ها (labels) را شاخص‌گذاری می‌کند و متن کامل لاگ را ذخیره می‌کند.

برای راه‌اندازی Loki، که یک سیستم ذخیره‌سازی و جستجوی لاگ متن باز است، می‌توانید مراحل زیر را دنبال کنید. این دستورالعمل‌ها برای سیستم عامل اوبونتو نوشته شده‌اند، اما با تغییرات جزئی می‌توان آنها را در سیستم‌های دیگر نیز استفاده کرد.

ابتدا باید فایل باینری Loki را دانلود و نصب کنید.

```
amir@amir:~$ curl -O -L "https://github.com/grafana/loki/releases/download/v2.9.0/loki-linux-amd64.zip"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 19.7M  100 19.7M    0     0   927k      0  0:00:21  0:00:21 --:--:-- 1135k
amir@amir:~$ unzip loki-linux-amd64.zip
Archive:  loki-linux-amd64.zip
  inflating: loki-linux-amd64
amir@amir:~$ sudo mv loki-linux-amd64 /usr/local/bin/loki
amir@amir:~$ sudo chmod +x /usr/local/bin/loki
```

شکل ۴۳-۳: دانلود و نصب Loki.

یک فایل پیکربندی برای Loki ایجاد کنید. این فایل تنظیمات Loki را مشخص می‌کند.

```
amir@amir:~$ sudo nano /etc/loki-local-config.yaml
```

شکل ۴۳-۳: ایجاد پیکربندی برای Loki.

فایل پیکربندی loki-local-config.yaml:

```
  GNU nano 7.2                          /etc/loki-local-config.yaml *
auth_enabled: false

server:
  http_listen_port: 3100

ingester:
  lifecycler:
    address: 127.0.0.1
    ring:
      kvstore:
        store: inmemory
      replication_factor: 1
    final_sleep: 0s
  chunk_idle_period: 3m
  chunk_retain_period: 1m
  max_transfer_retries: 0

schema_config:
  configs:
    - from: 2020-10-24
      store: boltdb-shipper
      object_store: filesystem
      schema: v11
      index:
        prefix: index_
        period: 24h

storage_config:
  boltdb_shipper:
    active_index_directory: /tmp/loki/boltdb-shipper-active
    cache_location: /tmp/loki/boltdb-shipper-cache
    shared_store: filesystem
  filesystem:
    directory: /tmp/loki/chunks

limits_config:
  enforce_metric_name: false
  reject_old_samples: true
  reject_old_samples_max_age: 168h

chunk_store_config:
  max_look_back_period: 0s

table_manager:
  retention_deletes_enabled: false
  retention_period: 0s
```

شکل ۴۴-۳: پیکربندی Loki.

برای اینکه Loki به عنوان یک سرویس سیستم شروع شود، یک فایل سرویس Systemd ایجاد کنید:

```
amir@amir:~$ sudo nano /etc/systemd/system/loki.service
```

شکل ۴۵-۳: تنظیم Loki به عنوان یک فایل سیستم.

فایل پیکربندی :

```
amir@amir: ~
  GNU nano 7.2                    /etc/systemd/system/loki.service
[Unit]
Description=Loki Log Aggregator
After=network.target

[Service]
ExecStart=/usr/local/bin/loki -config.file /etc/loki-local-config.yaml
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

شکل ۴۶-۳: پیکربندی Loki.

سرویس Loki را شروع کرده و آن را فعال کنید که با هر بار راه‌اندازی سیستم به صورت خودکار شروع شود:

```
amir@amir:~$ sudo systemctl daemon-reload
amir@amir:~$ sudo systemctl start loki
amir@amir:~$ sudo systemctl enable loki
Created symlink /etc/systemd/system/multi-user.target.wants/loki.service → /etc/systemd/system/loki.servi
ce.
```

شکل ۴۷-۳: تنظیم فعال سازی خودکار Loki.

می‌توانید وضعیت سرویس Loki را با دستور زیر بررسی کنید:

sudo systemctl status loki

این مورد برای زخیره و نمایش وضعیت فعال بودن ssl مدت زمان گواهی و اطلاعات گواهی استفاده می شود.

۹-۱-۳- تست صحت عملکرد:

قبل از شروع ساخت داشبورد ها از صحت عملکرد برنامه و پلاگین ها مطمئن می شویم:

صحت عملکرد node exporter:

```
amir@amir:~$ systemctl status node_exporter
● node_exporter.service - Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; preset>
     Active: active (running) since Sat 2024-07-13 21:06:52 +0330; 1h 13min ago
   Main PID: 1145 (node_exporter)
      Tasks: 9 (limit: 9400)
     Memory: 24.8M
        CPU: 22.436s
     CGroup: /system.slice/node_exporter.service
             └─1145 /usr/local/bin/node_exporter

Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.948>
Jul 13 21:06:56 amir node_exporter[1145]: level=info ts=2024-07-13T17:36:55.949>
lines 1-20/20 (END)
```

شکل ۴۸-۳: صحت عملکرد node exporter.

صحت عملکرد promatheus:

```
amir@amir:~$ sudo systemctl status prometheus
● prometheus.service - Prometheus
     Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; preset: e>
     Active: active (running) since Sat 2024-07-13 22:07:52 +0330; 13min ago
   Main PID: 4706 (prometheus)
      Tasks: 10 (limit: 9400)
     Memory: 34.0M
        CPU: 2.325s
     CGroup: /system.slice/prometheus.service
             └─4706 /usr/local/bin/prometheus --config.file /etc/prometheus/pro>

Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.888Z caller=head.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.888Z caller=head.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.894Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.894Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.894Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.895Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.895Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.895Z caller=main.>
Jul 13 22:07:52 amir prometheus[4706]: ts=2024-07-13T18:37:52.896Z caller=manag>
Jul 13 22:07:57 amir prometheus[4706]: ts=2024-07-13T18:37:57.896Z caller=scrap>
lines 1-20/20 (END)
```

شکل ۴۹-۳: صحت عملکرد promatheus.

صحت عملکرد blackbox exporter:

```
amir@amir:~$ sudo systemctl status blackbox_exporter
● blackbox_exporter.service - Blackbox Exporter
     Loaded: loaded (/etc/systemd/system/blackbox_exporter.service; enabled; pr>
     Active: active (running) since Sat 2024-07-13 21:06:52 +0330; 1h 16min ago
   Main PID: 1137 (blackbox_export)
      Tasks: 8 (limit: 9400)
     Memory: 14.2M
        CPU: 147ms
     CGroup: /system.slice/blackbox_exporter.service
             └─1137 /usr/local/bin/blackbox_exporter --config.file /etc/prometh>

Jul 13 21:06:52 amir systemd[1]: Started blackbox_exporter.service - Blackbox E>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=info ts=2024-07-13T17:36:55>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=info ts=2024-07-13T17:36:55>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=warn ts=2024-07-13T17:36:56>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=info ts=2024-07-13T17:36:56>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=info ts=2024-07-13T17:36:56>
Jul 13 21:06:56 amir blackbox_exporter[1137]: level=info ts=2024-07-13T17:36:56>
```

شکل ۵۰-۳: صحت عملکرد blackbox exporter.

صحت عملکرد Grafana:

```
amir@amir: ~

amir@amir:~$ sudo systemctl status grafana-server
● grafana-server.service - Grafana instance
     Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; prese>
     Active: active (running) since Sat 2024-07-13 21:06:52 +0330; 1h 18min ago
       Docs: http://docs.grafana.org
   Main PID: 1139 (grafana)
      Tasks: 15 (limit: 9400)
     Memory: 186.9M
        CPU: 16.807s
     CGroup: /system.slice/grafana-server.service
             └─1139 /usr/share/grafana/bin/grafana server --config=/etc/grafana>

Jul 13 21:31:15 amir grafana[1139]: logger=live t=2024-07-13T21:31:15.299682026>
Jul 13 21:37:24 amir grafana[1139]: logger=cleanup t=2024-07-13T21:37:24.289906>
Jul 13 21:38:57 amir grafana[1139]: logger=infra.usagestats t=2024-07-13T21:38:>
Jul 13 21:39:13 amir grafana[1139]: logger=live t=2024-07-13T21:39:13.652828497>
Jul 13 21:44:47 amir grafana[1139]: logger=live t=2024-07-13T21:44:47.308245141>
Jul 13 21:47:24 amir grafana[1139]: logger=cleanup t=2024-07-13T21:47:24.295018>
Jul 13 21:57:24 amir grafana[1139]: logger=cleanup t=2024-07-13T21:57:24.300690>
Jul 13 22:07:24 amir grafana[1139]: logger=cleanup t=2024-07-13T22:07:24.289549>
Jul 13 22:08:57 amir grafana[1139]: logger=infra.usagestats t=2024-07-13T22:08:>
Jul 13 22:17:24 amir grafana[1139]: logger=cleanup t=2024-07-13T22:17:24.292237>
```

شکل ۵۱-۳: صحت عملکرد Grafana.

صحت عملکرد InfluxDB:

```
amir@amir:~$ sudo systemctl status influxdb
● influxdb.service - InfluxDB is an open-source, distributed, time series database
     Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; preset: enabled)
     Active: active (running) since Mon 2024-07-15 16:27:20 +0330; 2min 27s ago
       Docs: man:influxd(1)
   Main PID: 9635 (influxd)
      Tasks: 9 (limit: 9400)
     Memory: 6.7M
        CPU: 107ms
     CGroup: /system.slice/influxdb.service
             └─9635 /usr/bin/influxd -config /etc/influxdb/influxdb.conf

Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745512Z lvl=info msg="Registered diagnos>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745538Z lvl=info msg="Starting precreati>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745553Z lvl=info msg="Starting snapshot >
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745561Z lvl=info msg="Starting continuou>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745573Z lvl=info msg="Starting HTTP serv>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745579Z lvl=info msg="opened HTTP access>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745667Z lvl=info msg="Storing statistics>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745698Z lvl=info msg="Listening on HTTP">
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.745722Z lvl=info msg="Starting retention>
Jul 15 16:27:20 amir influxd[9635]: ts=2024-07-15T12:57:20.746150Z lvl=info msg="Listening for sign>
```

شکل ۵۲-۳: صحت عملکرد InfluxDB.

۲-۳-نصب و راه اندازی ابزار ها بر روی widows

۱-۲-۳- دانلود و اجرا :

می توانیم از Prometheus و Grafana در ویندوز استفاده کنیم.

به این منظور ابتدا فایل های گرافانا و پرومتئوس ورژن ویندوز را از لیک های زیر دانلود می نماییم:

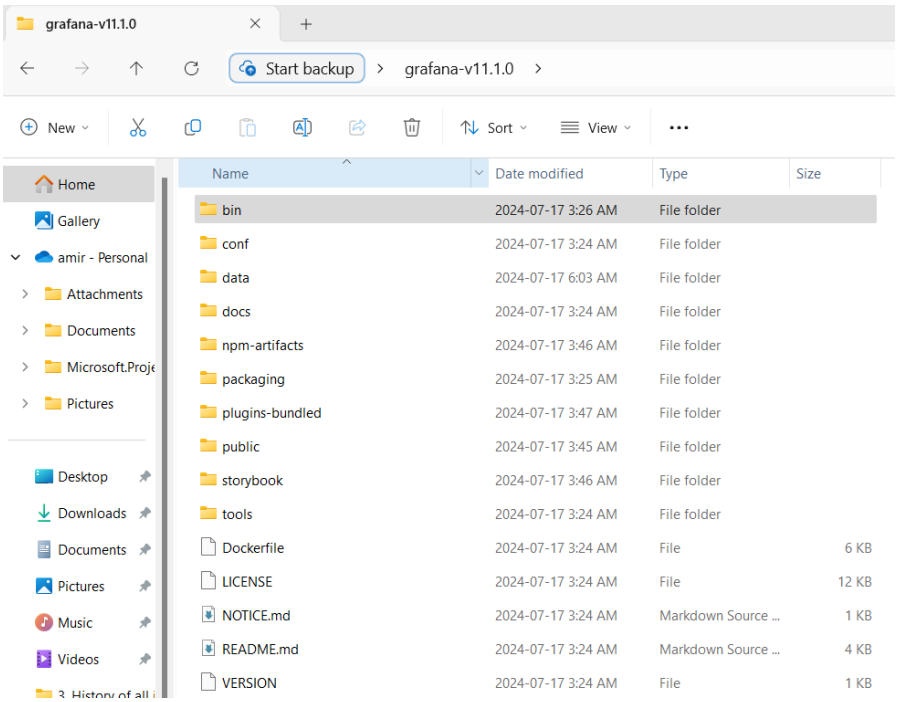
لینک دانلود پرومتئوس:

https://dl.grafana.com/enterprise/release/grafana-enterprise-۱۱.۱.۰.windows-amd۶۴.zip

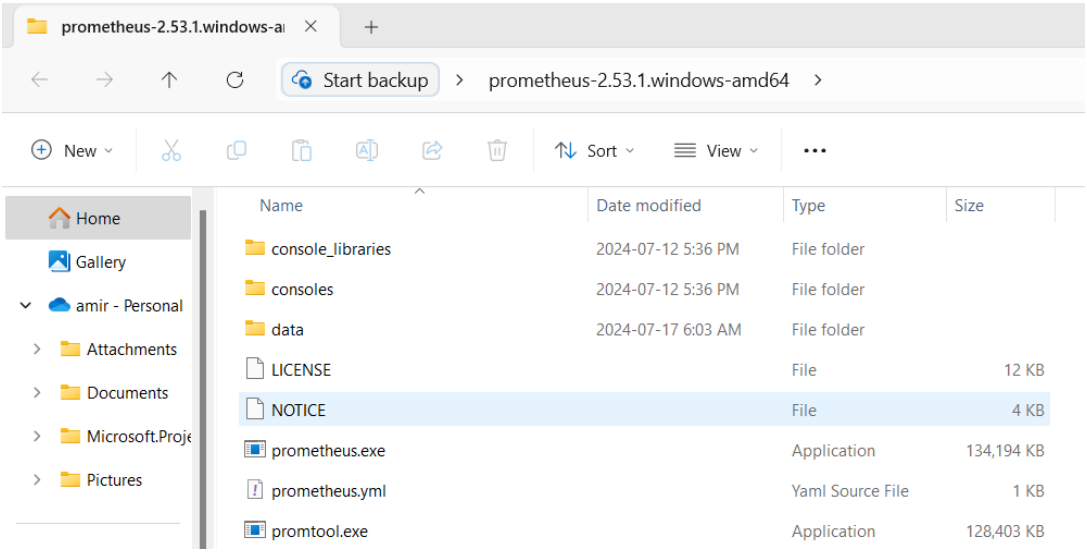
لینک دانلود گرافانا:

https://github.com/prometheus/prometheus/releases/download/v2.53.1/prometheus-2.53.1.windows-amd64.zip

سپس فایل های دانلود شده را آن زیپ میکنیم. (از حالت فشرده خارج می کنیم.)



شکل۵۳-۳: دانلود و آنزیپ Grafana.



شکل۵۴-۳: دانلود و آنزیپ Grafana.

سپس ابتدا از پوشه گرافانا از دایرکتوری بین فایل grafana-server.exe را اجرا می کنیم.

سپس از پوشه پرومتئوس فایل prometheus.exe را اجرا می کنیم تا اتصال برقرار شود.

۲-۲-۳- دانلود و اجرا :

تصویر صحت اجرا:

صحت اجرای گرافانا:

```
C:\Users\IP3\Desktop\grafana
INFO [07-17|23:50:01] Plugins loaded                          logger=plugin.store count=54 duration=639.0212ms
INFO [07-17|23:50:01] Caching service is disabled             logger=caching.service
INFO [07-17|23:50:02] Query Service initialization            logger=query_data
INFO [07-17|23:50:02] Live Push Gateway initialization        logger=live.push_http
INFO [07-17|23:50:02] Applying new configuration to Alertmanager logger=ngalert.notifier.alertmanager org=1 configHash=d
2c56faca6af2a5772ff4253222f7386
INFO [07-17|23:50:02] Running in alternative execution of Error/NoData mode logger=ngalert.state.manager
INFO [07-17|23:50:02] registering usage stat providers        logger=infra.usagestats.collector usageStatsProvidersLen=
3
INFO [07-17|23:50:02] starting to provision alerting          logger=provisioning.alerting
INFO [07-17|23:50:02] finished to provision alerting          logger=provisioning.alerting
ERROR[07-17|23:50:02] Failed to get renderer plugin sources   logger=renderer.manager error="failed to open plugins pat
h"
INFO [07-17|23:50:02] Warming state cache for startup         logger=ngalert.state.manager
INFO [07-17|23:50:02] Storage starting                        logger=grafanaStorageLogger
WARN [07-17|23:50:02] Scheduling and sending of reports disabled, SMTP is not configured and enabled. Configure SMTP to
enable. logger=report
INFO [07-17|23:50:02] Starting MultiOrg Alertmanager          logger=ngalert.multiorg.alertmanager
INFO [07-17|23:50:02] HTTP Server Listen                      logger=http.server address=[::]:3000 protocol=http subUrl
= socket=
INFO [07-17|23:50:02] starting to provision dashboards        logger=provisioning.dashboard
INFO [07-17|23:50:02] finished to provision dashboards        logger=provisioning.dashboard
INFO [07-17|23:50:02] State cache has been initialized        logger=ngalert.state.manager states=0 duration=106.3782ms
INFO [07-17|23:50:02] Starting scheduler                      logger=ngalert.scheduler tickInterval=10s maxAttempts=1
INFO [07-17|23:50:02] starting                                logger=ticker first_tick=2024-07-17T23:50:10+03:30
INFO [07-17|23:50:02] Adding GroupVersion playlist.grafana.app v0alpha1 to ResourceManager logger=grafana-apiserver
INFO [07-17|23:50:02] Adding GroupVersion featuretoggle.grafana.app v0alpha1 to ResourceManager logger=grafana-apiserver
INFO [07-17|23:50:02] Update check succeeded                  logger=plugins.update.checker duration=395.2834ms
INFO [07-17|23:50:02] Update check succeeded                  logger=grafana.update.checker duration=396.3203ms
```

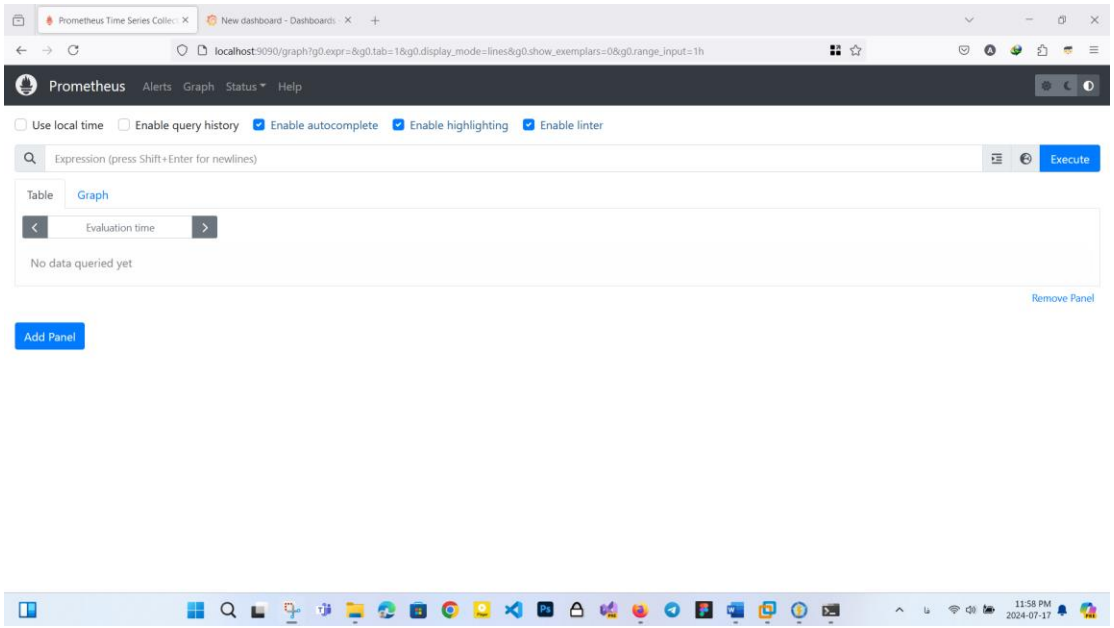
شکل ۵۵-۳: صحت اجرای Grafana.

صحت اجرای پرومتئوس:

```
C:\Users\IP3\Desktop\promet
ts=2024-07-17T20:20:15.720Z caller=head.go:793 level=info component=tsdb msg="WAL segment loaded" segment=8 maxSegment=8
ts=2024-07-17T20:20:15.720Z caller=head.go:830 level=info component=tsdb msg="WAL replay completed" checkpoint_replay_du
ration=11.0841ms wal_replay_duration=37.3674ms wbl_replay_duration=0s chunk_snapshot_load_duration=0s mmap_chunk_replay_
duration=5.0533ms total_replay_duration=54.0103ms
ts=2024-07-17T20:20:15.723Z caller=main.go:1169 level=info fs_type=unknown
ts=2024-07-17T20:20:15.723Z caller=main.go:1172 level=info msg="TSDB started"
ts=2024-07-17T20:20:15.723Z caller=main.go:1354 level=info msg="Loading configuration file" filename=prometheus.yml
ts=2024-07-17T20:20:16.623Z caller=main.go:1391 level=info msg="updated GOGC" old=100 new=75
ts=2024-07-17T20:20:16.623Z caller=main.go:1402 level=info msg="Completed loading of configuration file" filename=promet
heus.yml totalDuration=900.8505ms db_storage=0s remote_storage=0s web_handler=0s query_engine=0s scrape=899.3374ms scrap
e_sd=0s notify=0s notify_sd=0s rules=0s tracing=0s
ts=2024-07-17T20:20:16.624Z caller=main.go:1133 level=info msg="Server is ready to receive web requests."
ts=2024-07-17T20:20:16.624Z caller=manager.go:164 level=info component="rule manager" msg="Starting rule manager..."
ts=2024-07-17T20:20:30.098Z caller=compact.go:567 level=info component=tsdb msg="write block resulted in empty block" mi
nt=1720908000000 maxt=1720915200000 duration=30.4319ms
ts=2024-07-17T20:20:30.102Z caller=head.go:1355 level=info component=tsdb msg="Head GC completed" caller=truncateMemory
duration=3.8258ms
ts=2024-07-17T20:20:30.169Z caller=compact.go:576 level=info component=tsdb msg="write block" mint=1721175870050 maxt=17
21181600000 ulid=01J315N0RPAZA8D245FRJCJ6FK duration=66.5997ms ooo=false
ts=2024-07-17T20:20:30.173Z caller=head.go:1355 level=info component=tsdb msg="Head GC completed" caller=truncateMemory
duration=2.2268ms
ts=2024-07-17T20:20:30.239Z caller=compact.go:576 level=info component=tsdb msg="write block" mint=1721183607058 maxt=17
21188800000 ulid=01J315N0TYV4DDZ24EFD1MSPC8 duration=64.8512ms ooo=false
ts=2024-07-17T20:20:30.243Z caller=head.go:1355 level=info component=tsdb msg="Head GC completed" caller=truncateMemory
duration=2.5201ms
ts=2024-07-17T20:20:30.244Z caller=checkpoint.go:101 level=info component=tsdb msg="Creating checkpoint" from_segment=4
to_segment=6 mint=1721188800000
ts=2024-07-17T20:20:30.261Z caller=head.go:1317 level=info component=tsdb msg="WAL checkpoint complete" first=4 last=6 d
uration=18.4567ms
```

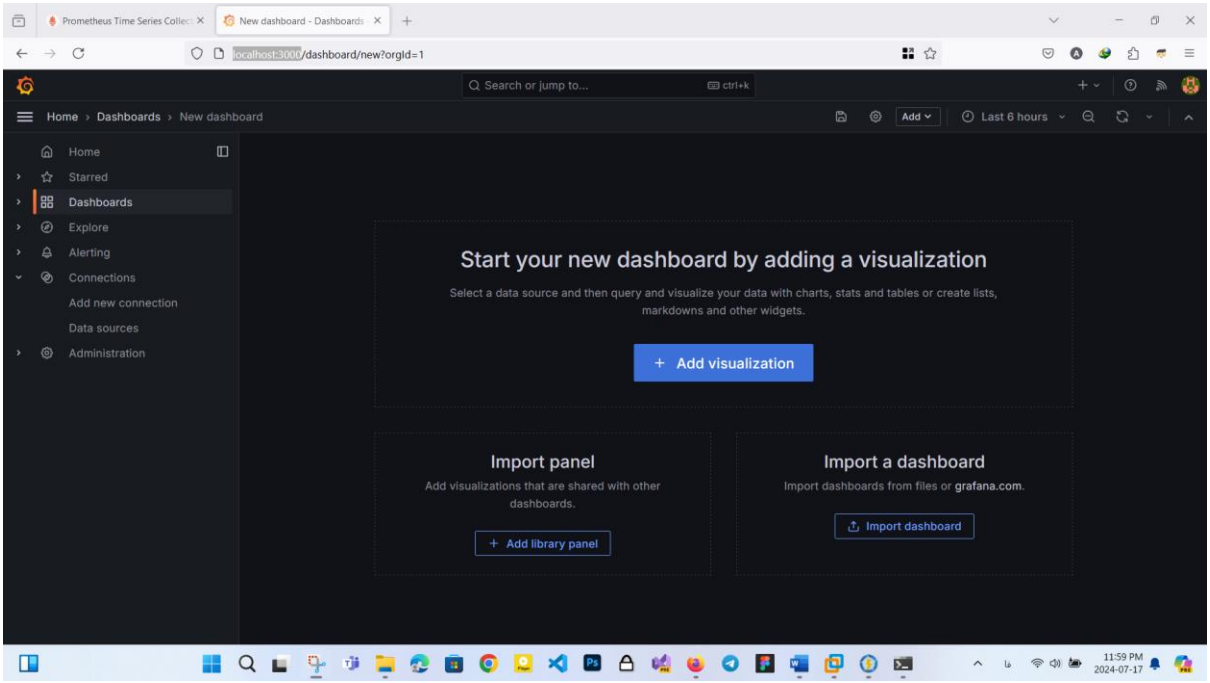
شکل ۵۶-۳: صحت اجرای Prometheus.

حال از طریق http://localhost:9090 می توانیم به پرومتئوس دسترسی دسته باشیم:



شکل ۵۷-۳: اجرای محیط Prometheus در ویندوز.

حال از طریق http://localhost:۳۰۰۰ می توانیم به گرافانا دسترسی دسته باشیم:



شکل ۵۸-۳: اجرای محیط Grafana در ویندوز.

حال میتوانیم دستورات promQL را در Prometheus اجرا کنیم و خروجی را در گرافانا مانیتور کنیم.

توجه: در این پروژه از Prometheus و Grafana در لینوکس (ubuntu) استفاده می کنیم. زیرا برخی دستورات نیاز به اجرا با اسکریپت های لینوکسی دارند زیرا پرومتئوس بطور معمول امکان دسترسی به موارد خواسته شده را ندارد. و در آینده در گزارش به آنها خواهیم پرداخت.

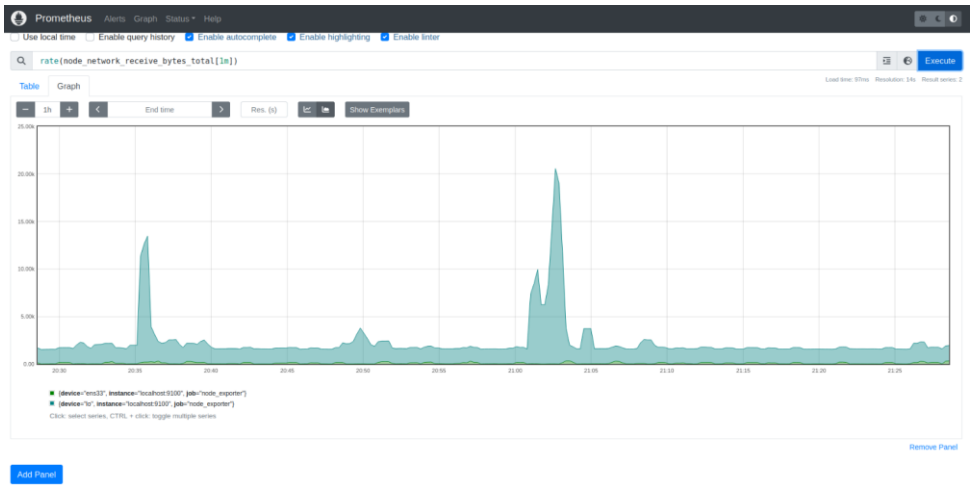
فصل چهارم

پیاده داشبورد ها با استفاده از Prometheus و Grafana

۱-۴-مانیتورینگ شبکه

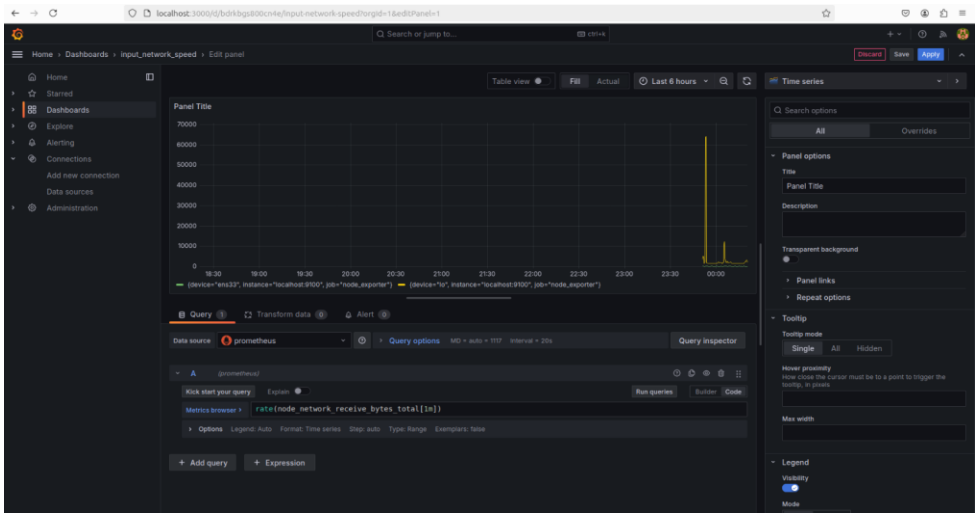
۱-۱-۴- سرعت شبکه ورودی برای همه کارتهای شبکه:

خروجی در Prometheus :



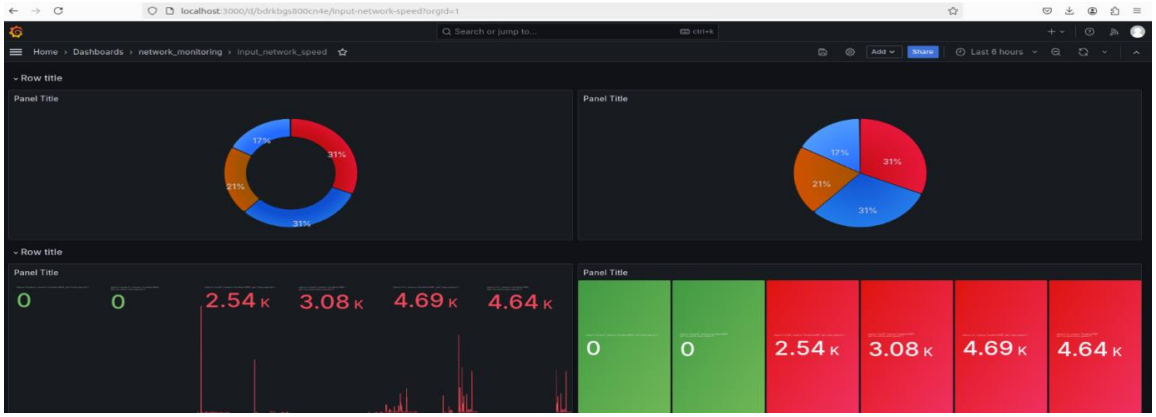
شکل ۱-۴: سرعت شبکه ورودی برای همه کارتهای شبکه (Prometheus).

خروجی در Grafana :



شکل ۲-۴: سرعت شبکه ورودی برای همه کارتهای شبکه (Grafana).

خروجی نهایی داشبورد:

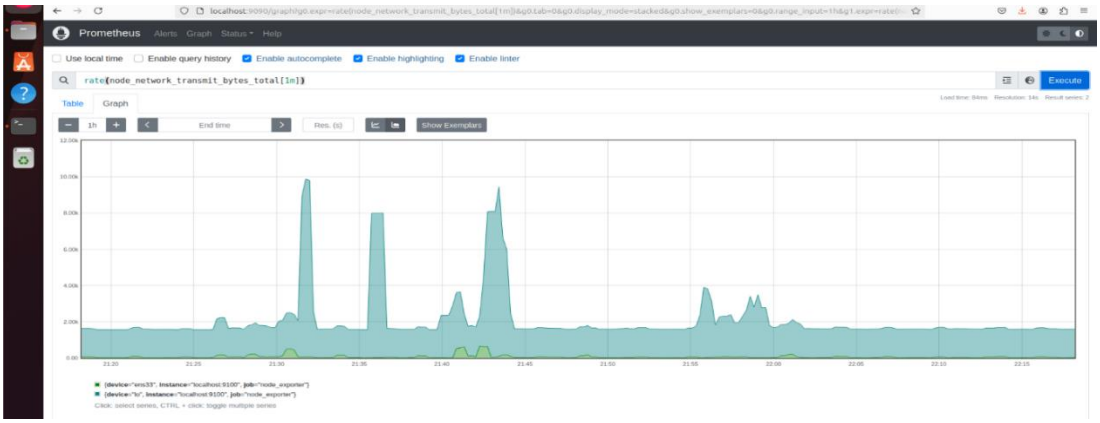


شکل ۳-۴: تصویر داشبورد گرافیکی کامل سرعت شبکه ورودی برای همه کارتهای شبکه (Grafana).

شکل ۴-۴: تصویر داشبورد گرافیکی کامل سرعت شبکه ورودی برای همه کارتهای شبکه (Grafana).

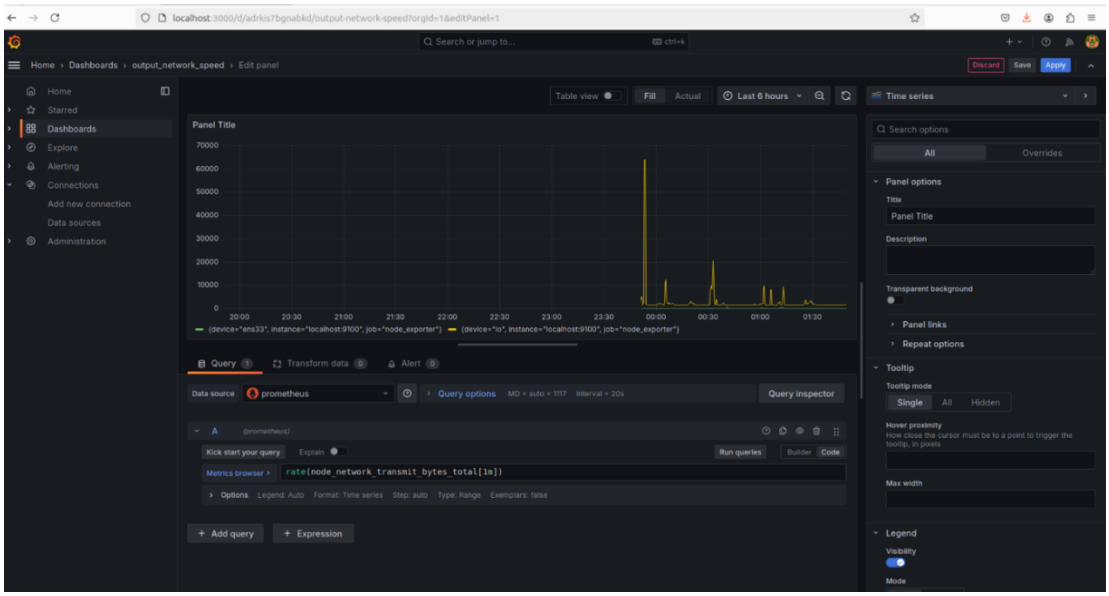
۲-۱-۴- سرعت شبکه خروجی برای همه کارتهای شبکه:

خروجی در Prometheus :



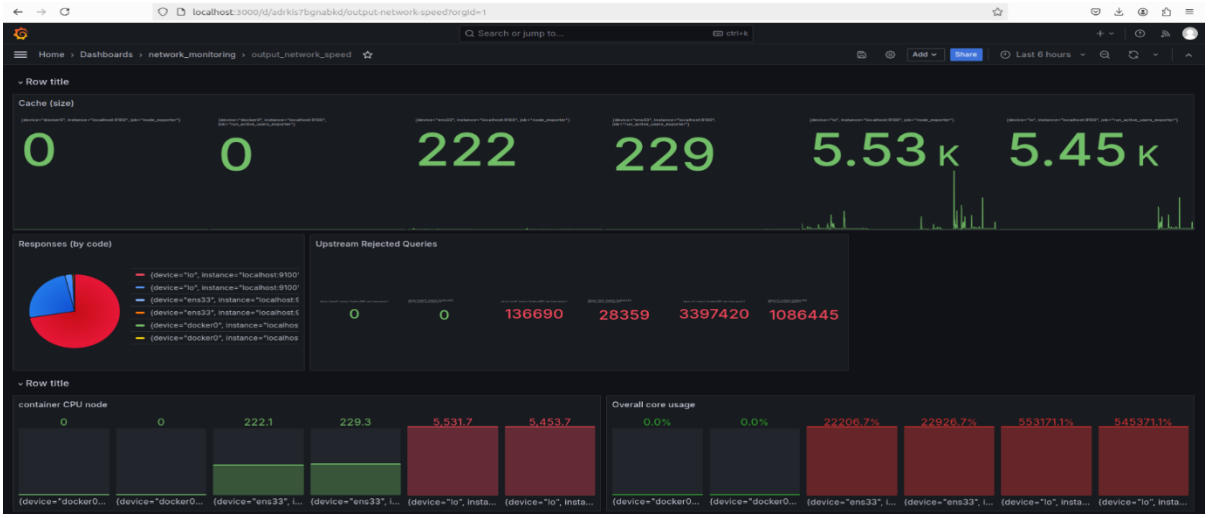
شکل ۵-۴: سرعت شبکه خروجی برای همه کارتهای شبکه (Prometheus).

خروجی در Grafana :

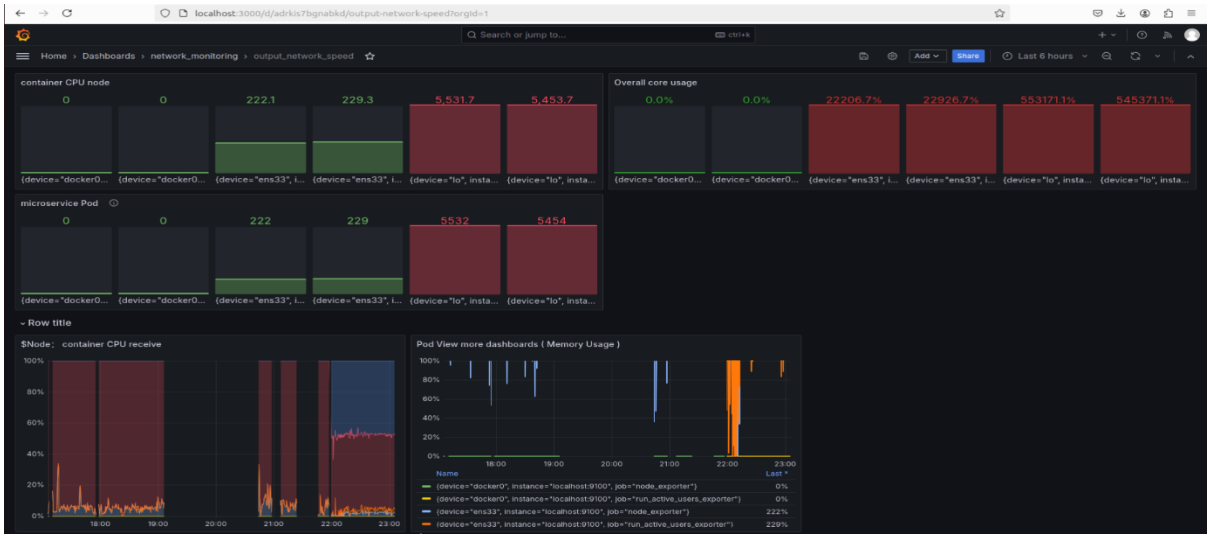


شکل ۶-۴: سرعت شبکه خروجی برای همه کارتهای شبکه (Grafana).

خروجی نهایی داشبورد:



شکل ۷-۴: تصویر داشبورد گرافیکی کامل سرعت شبکه خروجی برای همه کارتهای شبکه (Grafana).

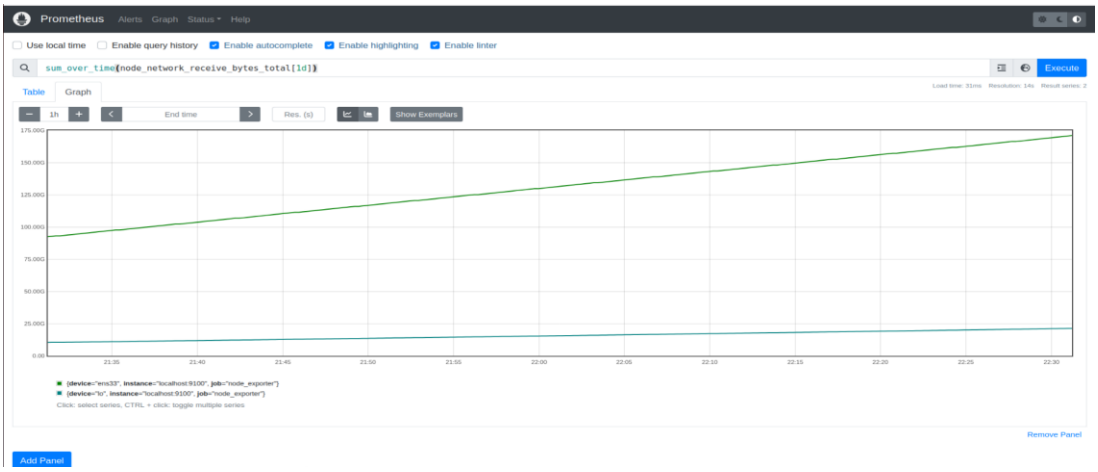


شکل ۸-۴: تصویر داشبورد گرافیکی کامل سرعت شبکه خروجی برای همه کارتهای شبکه (Grafana).

۳-۱-۴- هیستوری از کل ترافیک شبکه:

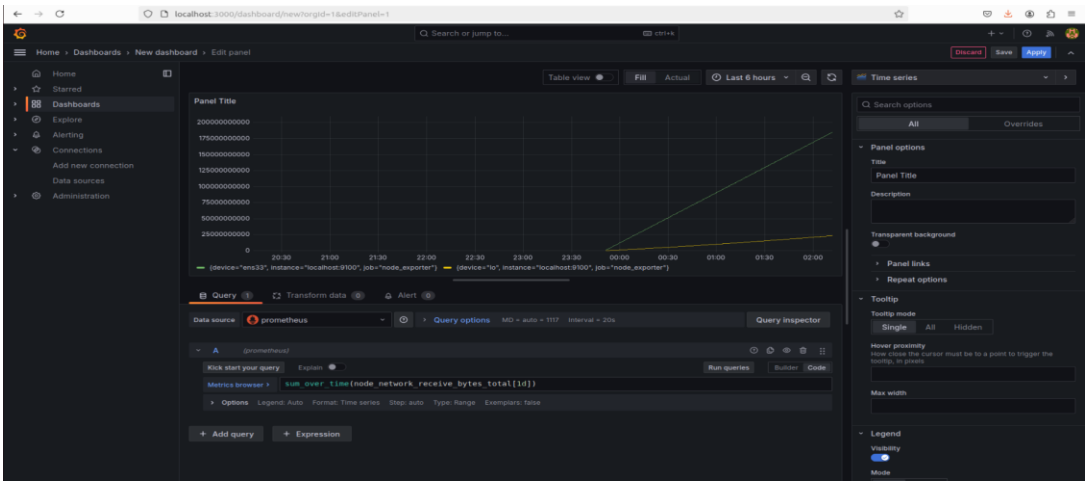
روزانه:

خروجی در Prometheus:



شکل ۹-۴: هیستوری از کل ترافیک شبکه روزانه (Prometheus).

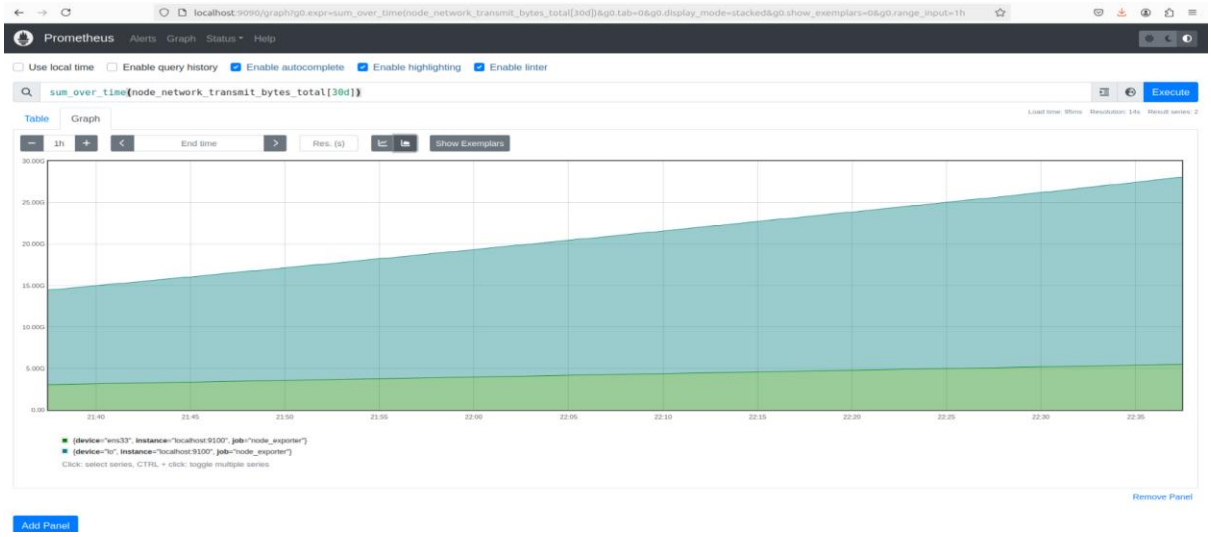
خروجی در Grafana:



شکل ۱۰-۴: هیستوری از کل ترافیک شبکه روزانه (Grafana).

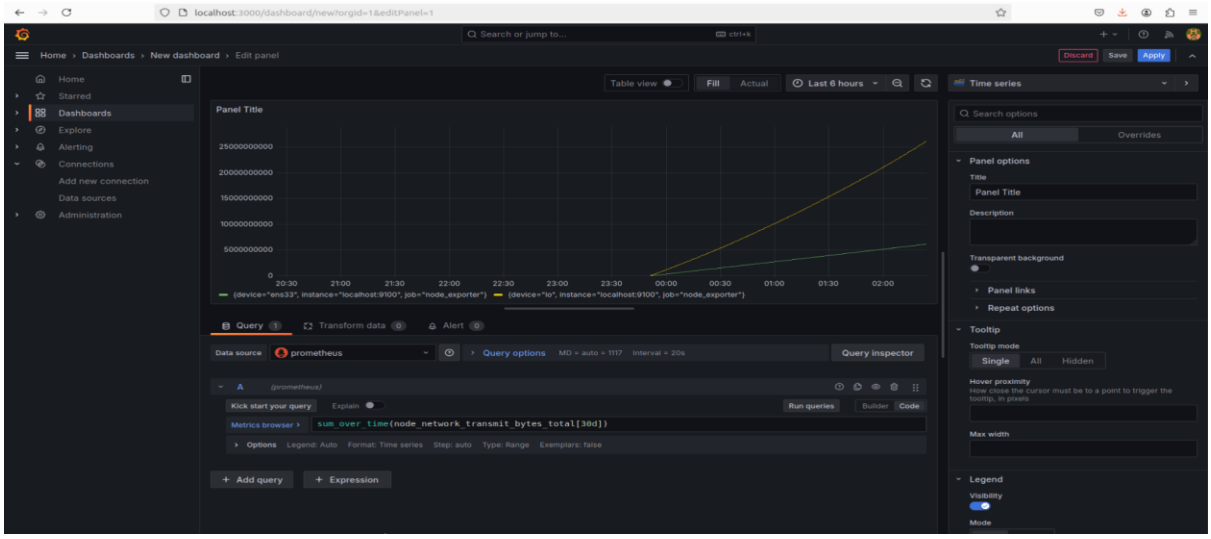
ماهانه:

خروجی در Prometheus :



شکل ۱۱-۴: هیستوری از کل ترافیک شبکه ماهانه (Prometheus).

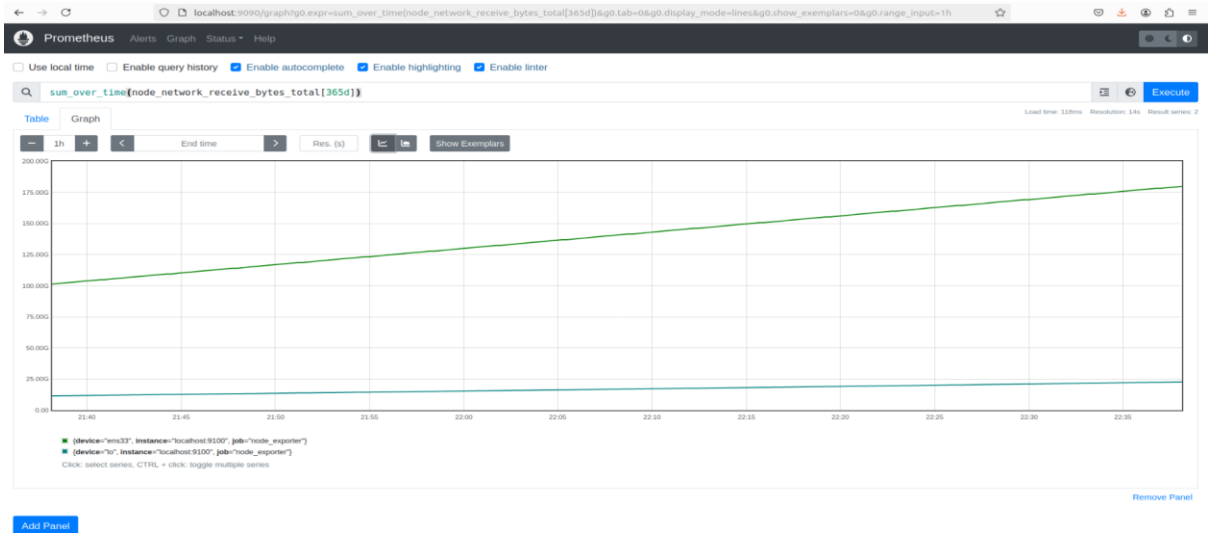
خروجی در Grafana :



شکل ۱۲-۴: هیستوری از کل ترافیک شبکه ماهانه (Grafana).

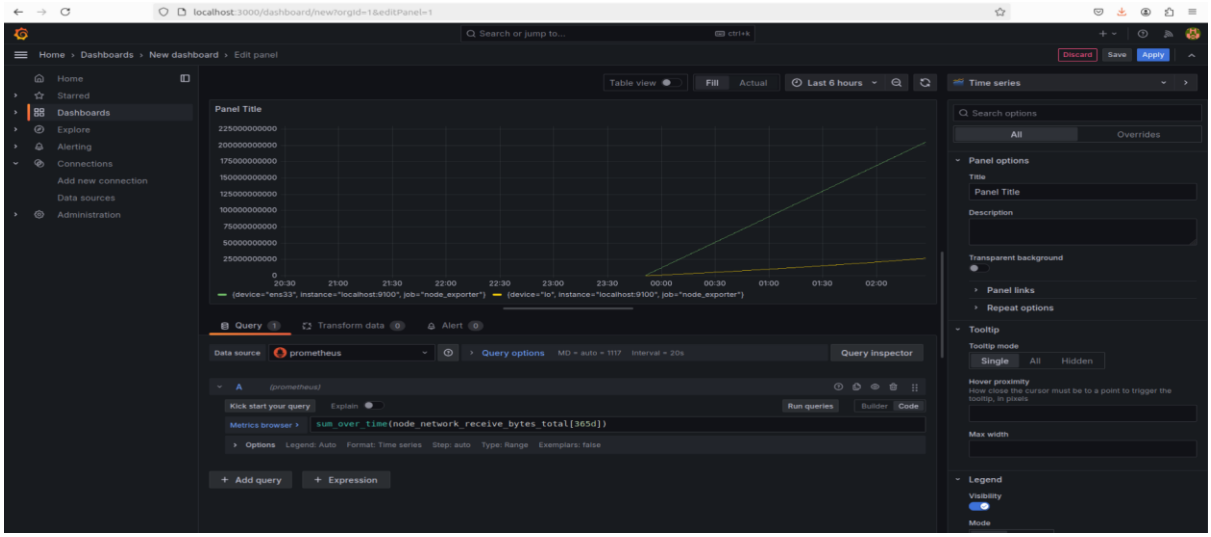
سالانه:

خروجی در Prometheus :



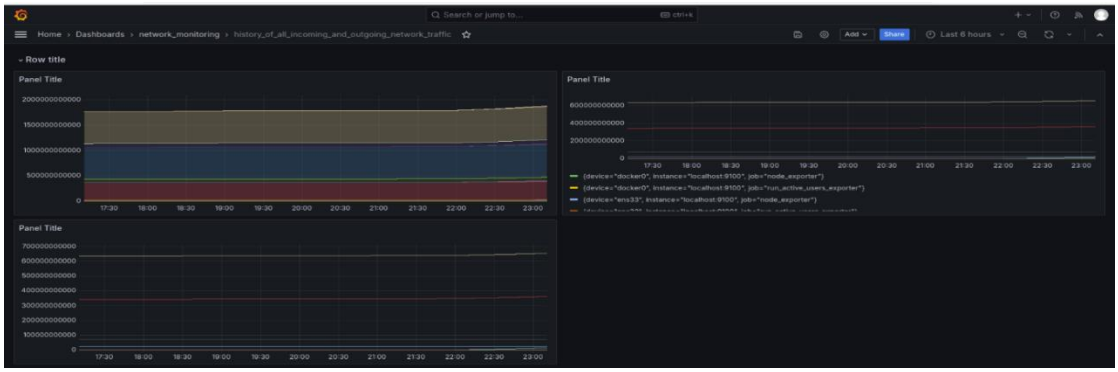
شکل ۱۳-۴: هیستوری از کل ترافیک شبکه سالانه (Prometheus).

خروجی در Grafana :

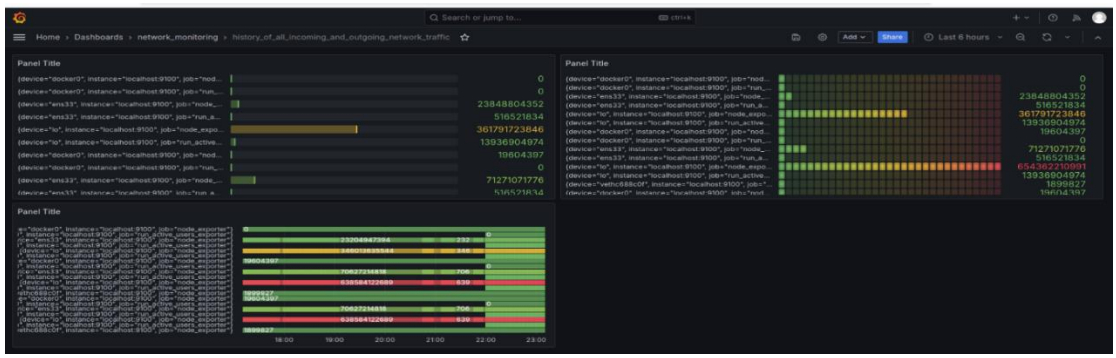


شکل ۱۴-۴: هیستوری از کل ترافیک شبکه سالانه (Grafana).

خروجی نهایی داشبورد:



شکل ۱۵-۴: تصویر داشبورد گرافیکی هیستوری از کل ترافیک شبکه (Grafana).



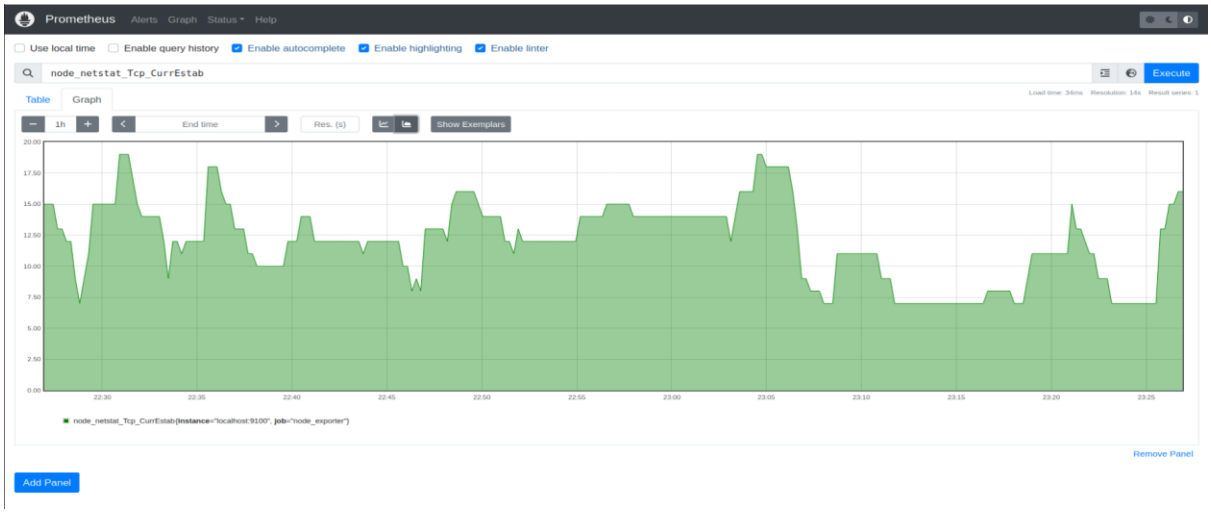
شکل ۱۶-۴: تصویر داشبورد گرافیکی هیستوری از کل ترافیک شبکه (Grafana).

شکل ۱۷-۴: تصویر داشبورد گرافیکی هیستوری از کل ترافیک شبکه (Grafana).

۴-۱-۴- نمایش کانکشن ها در لحظه:

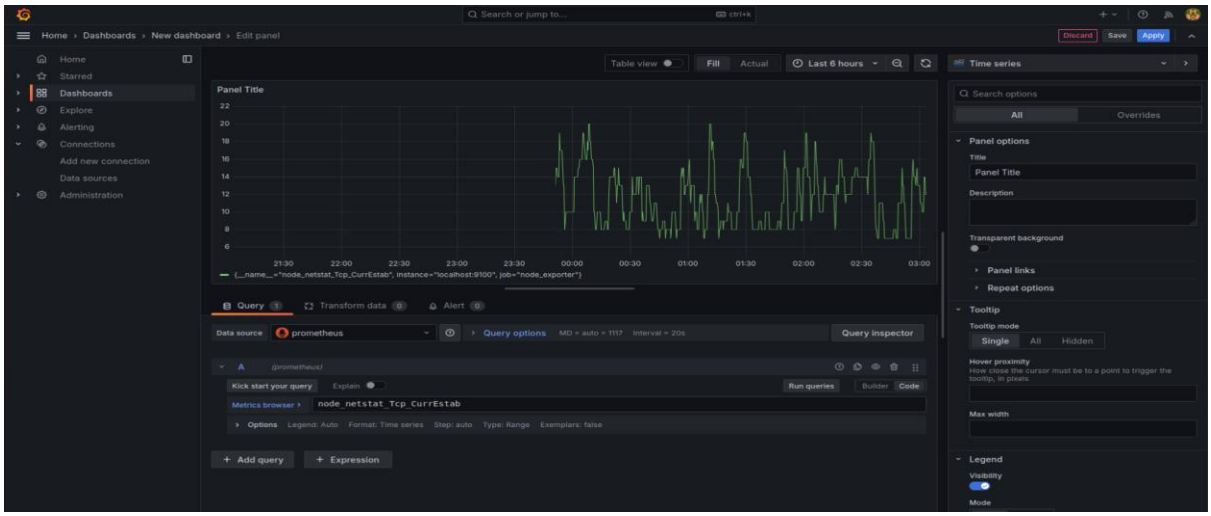
node_netstat_Tcp_CurrEstab(تعداد اتصالات)

خروجی در Prometheus:



شکل ۱۸-۴: نمایش کانکشن ها در لحظه (Prometheus).

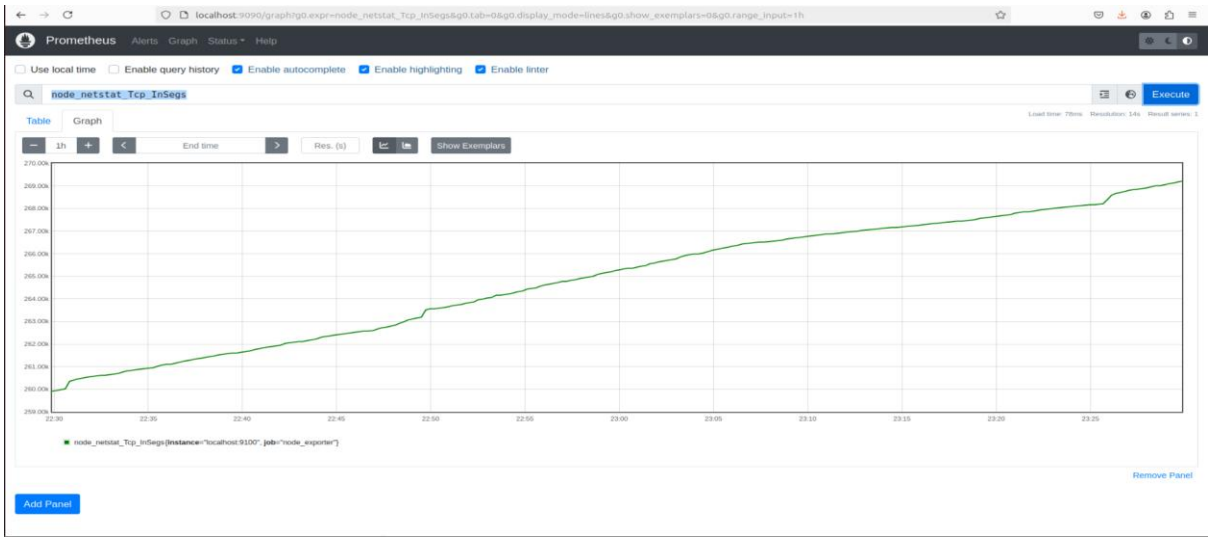
خروجی در Grafana:



شکل ۱۹-۴: نمایش کانکشن ها در لحظه (Grafana).

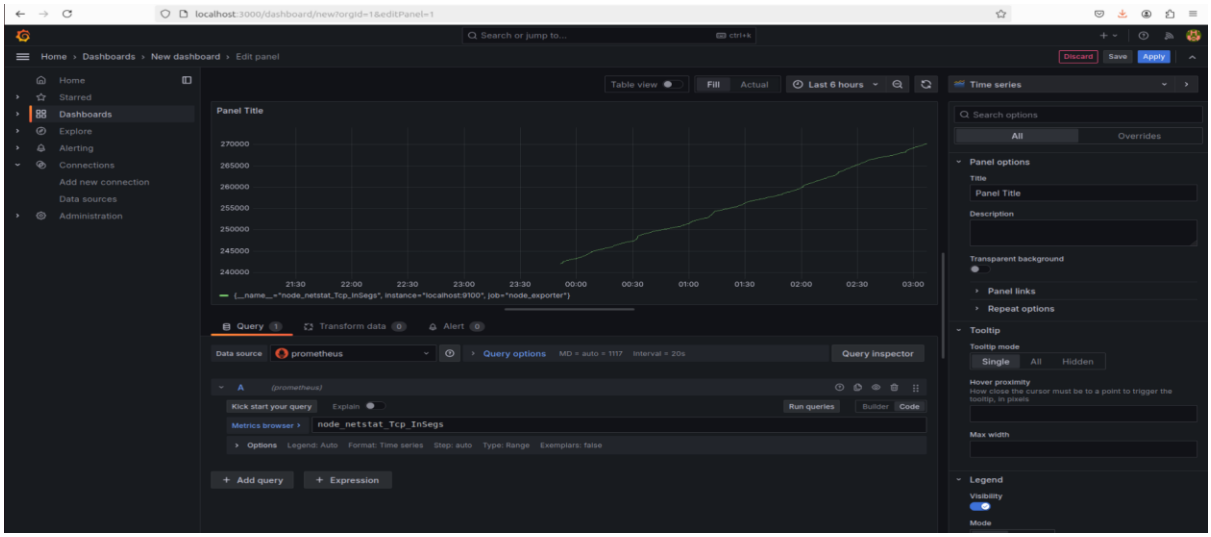
(تعداد سگمنت های دریافتی) node_netstat_Tcp_InSegs

خروجی در Prometheus :



شکل ۲۰-۴: نمایش کانکشن ها در لحظه (Prometheus).

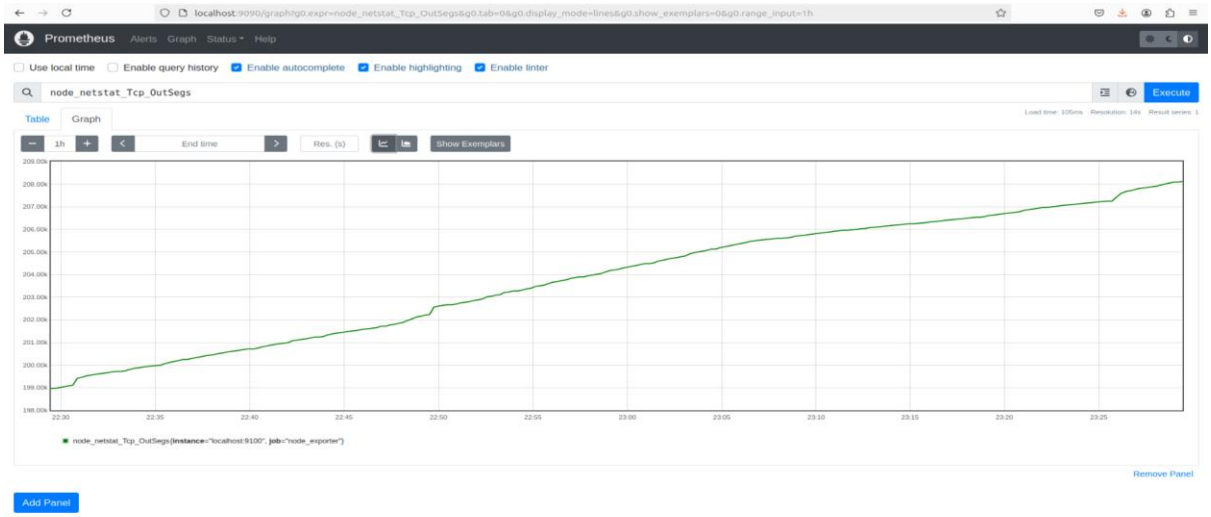
خروجی در Grafana :



شکل ۲۱-۴: نمایش کانکشن ها در لحظه (Grafana).

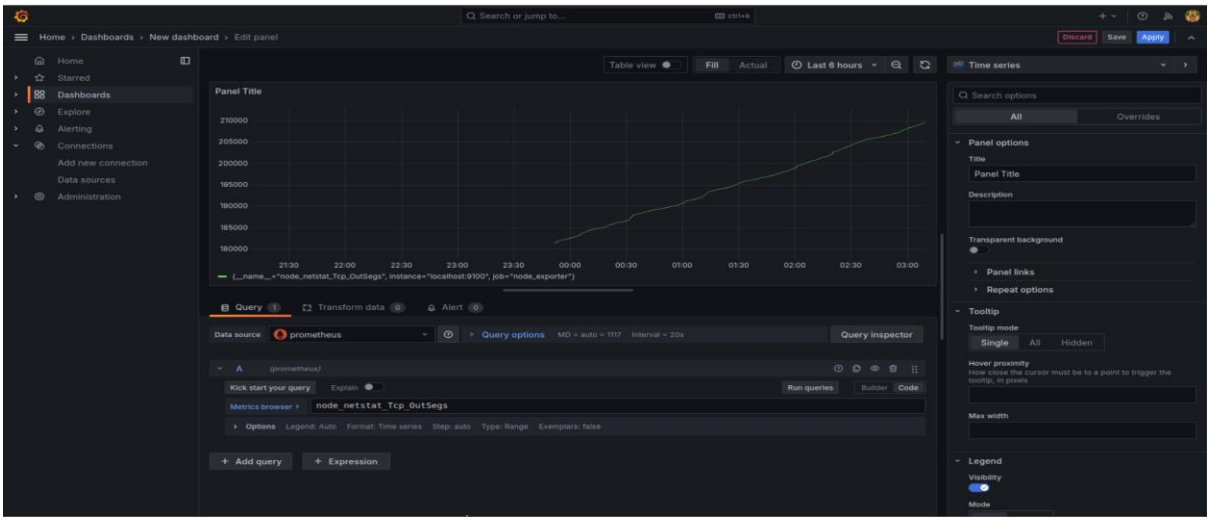
node_netstat_Tcp_OutSegs (تعداد سگمنت های ارسال شده)

خروجی در Prometheus :



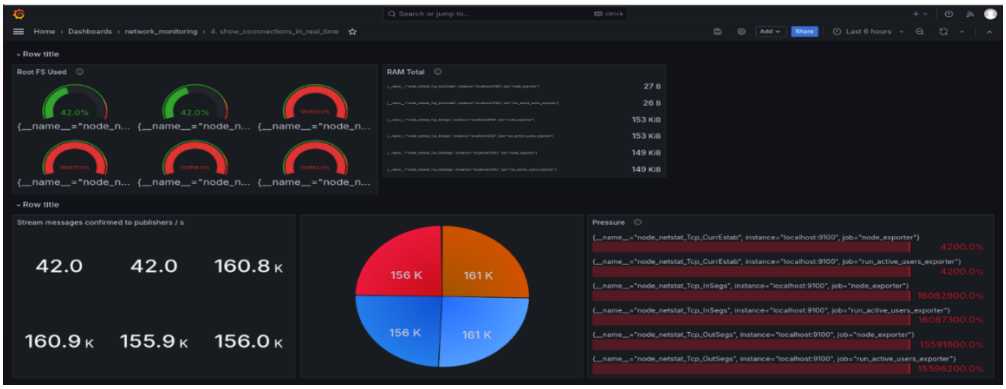
شکل ۲۲-۴: نمایش کانکشن ها در لحظه (Prometheus).

خروجی در Grafana :

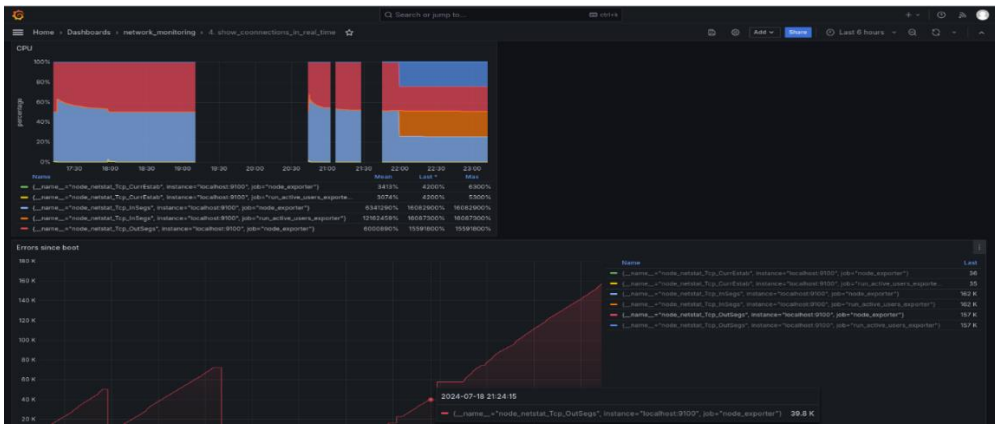


شکل ۲۳-۴: نمایش کانکشن ها در لحظه (Grafana).

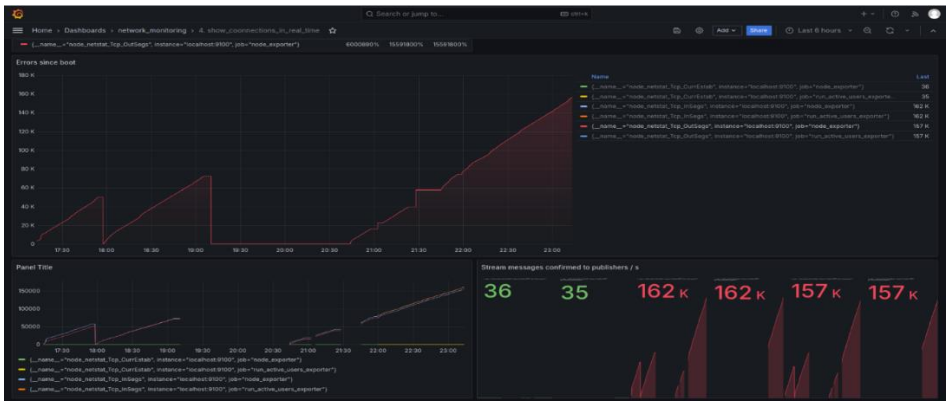
خروجی نهایی داشبورد:



شکل ۲۴-۴: تصویر کامل داشبورد گرافیکی نمایش کانکشن ها در لحظه (Grafana).



شکل ۲۵-۴: تصویر کامل داشبورد گرافیکی نمایش کانکشن ها در لحظه (Grafana).

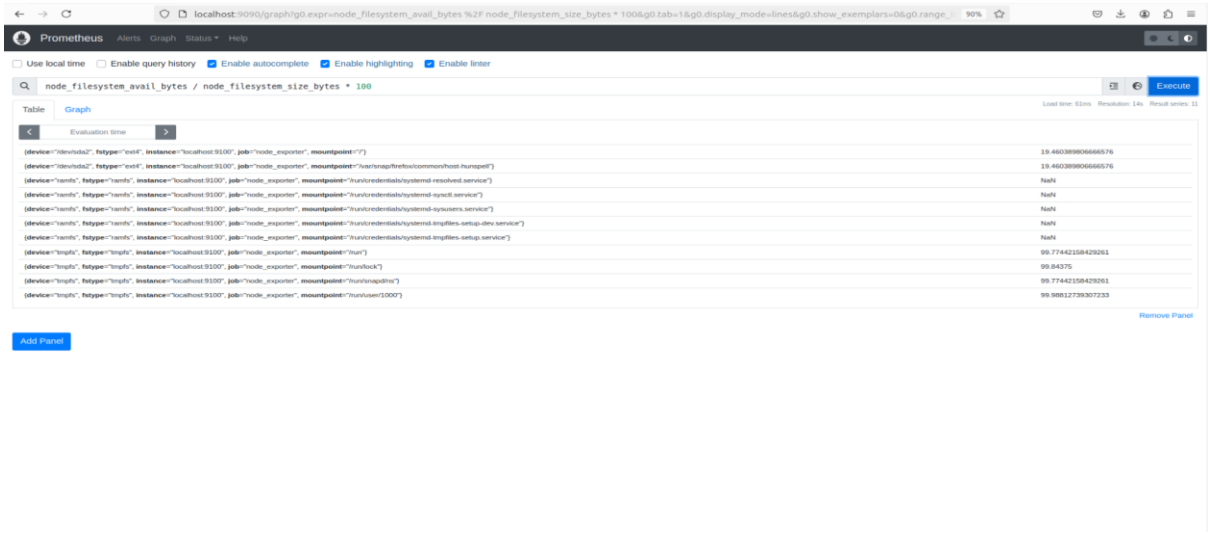


شکل ۲۶-۴: تصویر کامل داشبورد گرافیکی نمایش کانکشن ها در لحظه (Grafana).

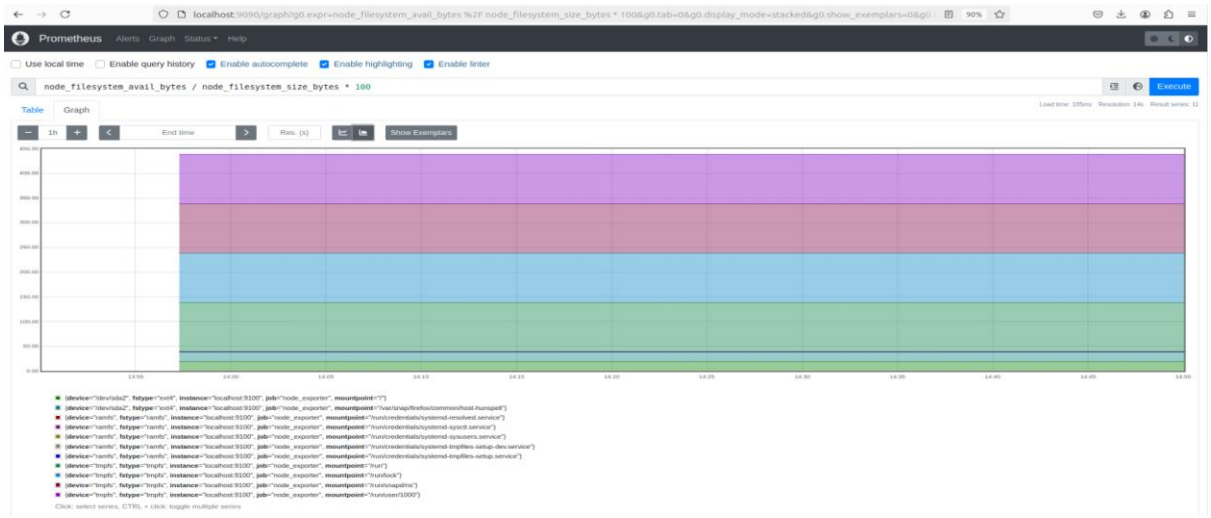
۲-۴- مانیتورینگ سیستم

۱-۲-۴- خلاصه وضعیت سخت افزار (دیسک) :

خروجی در Prometheus :

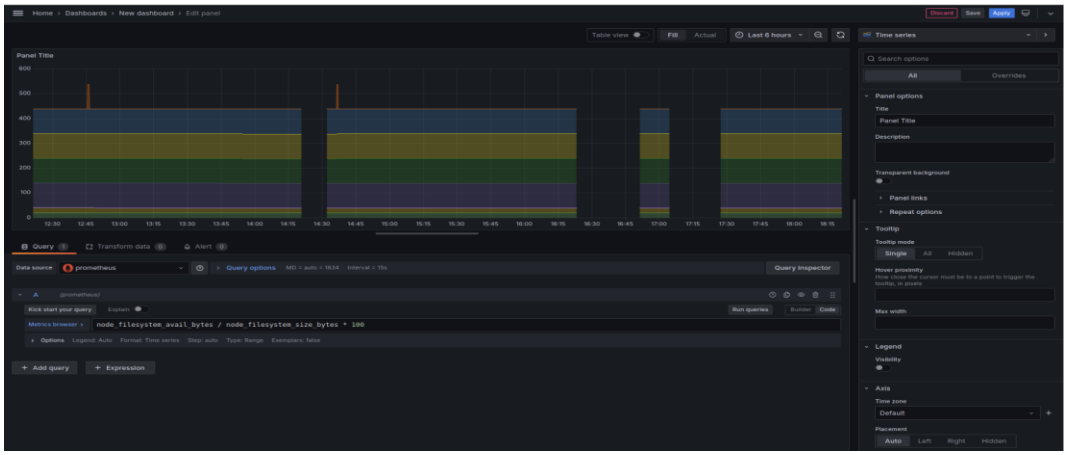


شکل ۲۷-۴: خلاصه وضعیت سخت افزار دیسک (Prometheus).

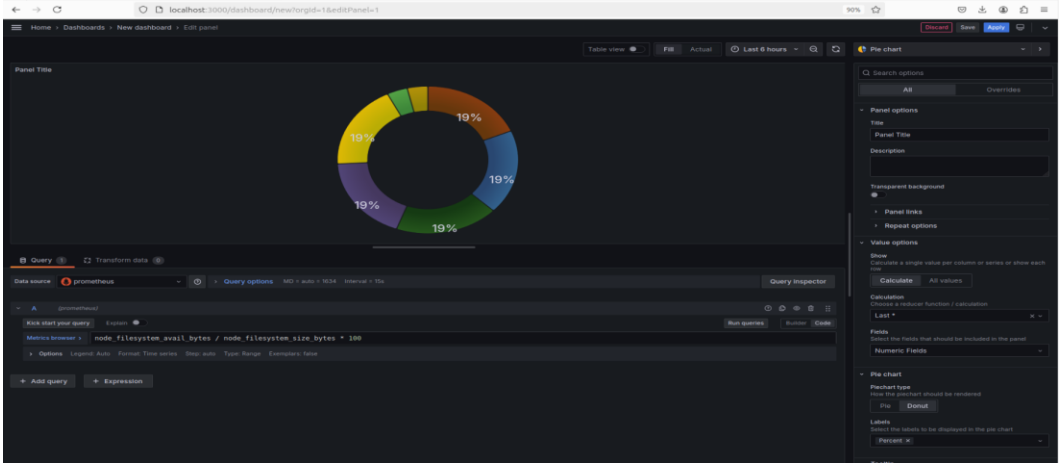


شکل ۲۸-۴: خلاصه وضعیت سخت افزار دیسک (Prometheus).

خروجی در Grafana :



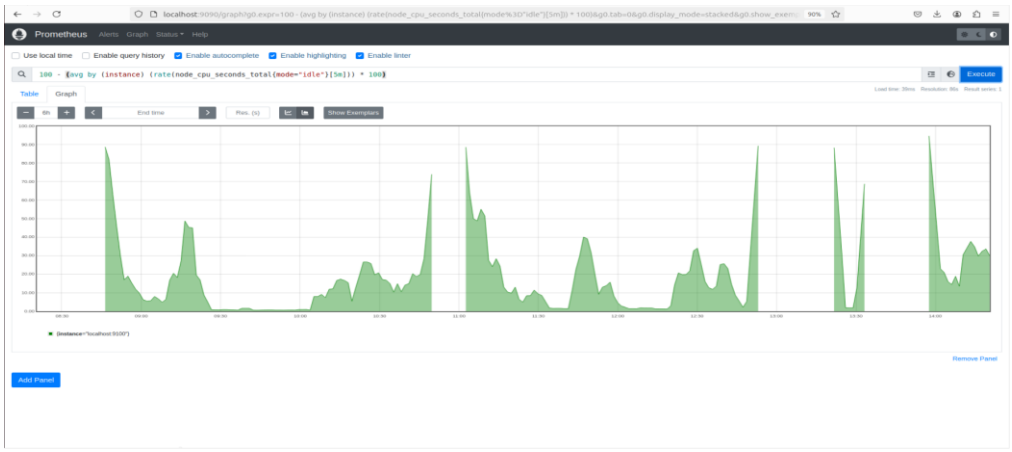
شکل ۲۹-۴: خلاصه وضعیت سخت افزار دیسک (Grafana).



شکل ۳۰-۴: خلاصه وضعیت سخت افزار دیسک (Grafana).

خلاصه وضعیت سخت افزار (نرخ بیکاری CPU) :

خروجی در Prometheus :



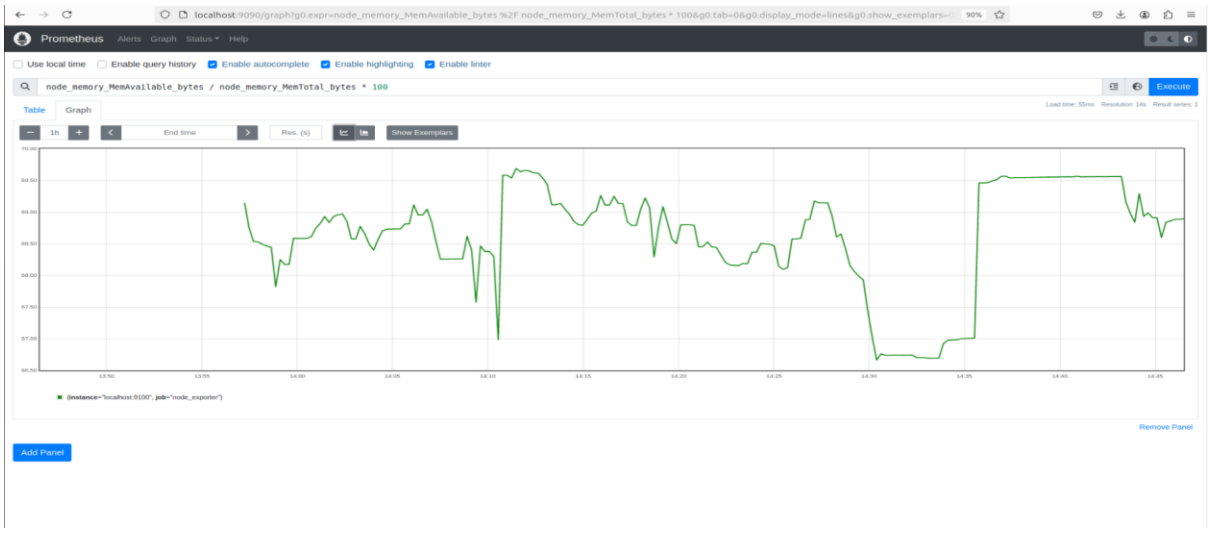
شکل ۳۱-۴: خلاصه وضعیت سخت افزار CPU (Prometheus).

خروجی در Grafana :

شکل ۳۲-۴: خلاصه وضعیت سخت افزار CPU (Grafana).

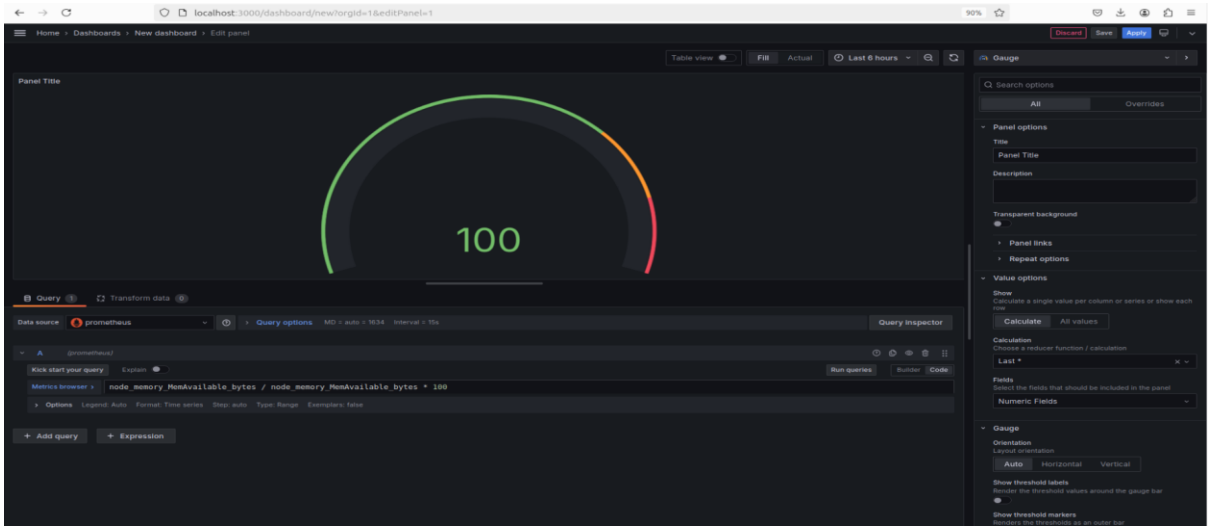
خلاصه وضعیت سخت افزار (حافظه) :

خروجی در Prometheus :



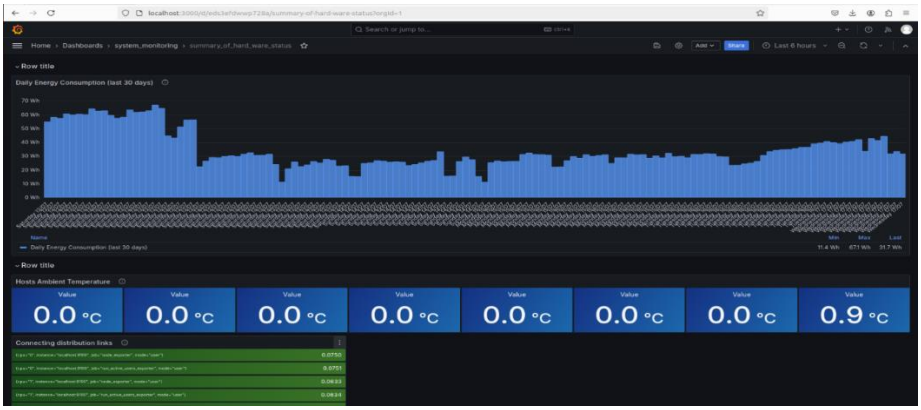
شکل ۳۳-۴: خلاصه وضعیت سخت افزار حافظه (Prometheus).

خروجی در Grafana :

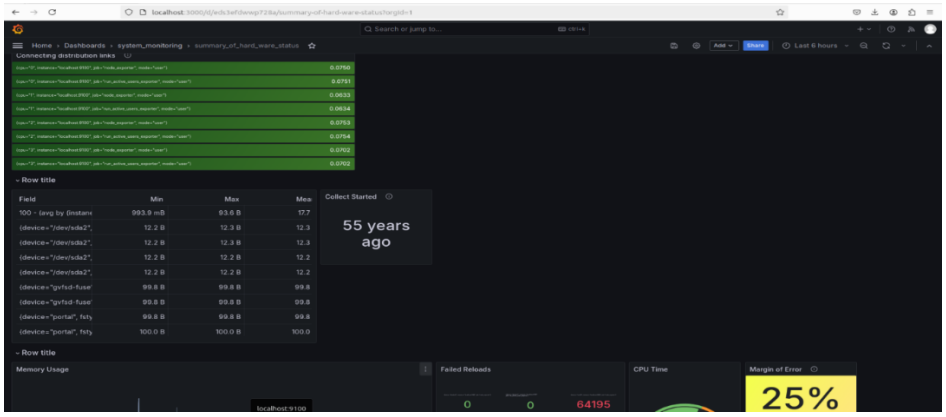


شکل ۳۴-۴: خلاصه وضعیت سخت افزار حافظه (Grafana).

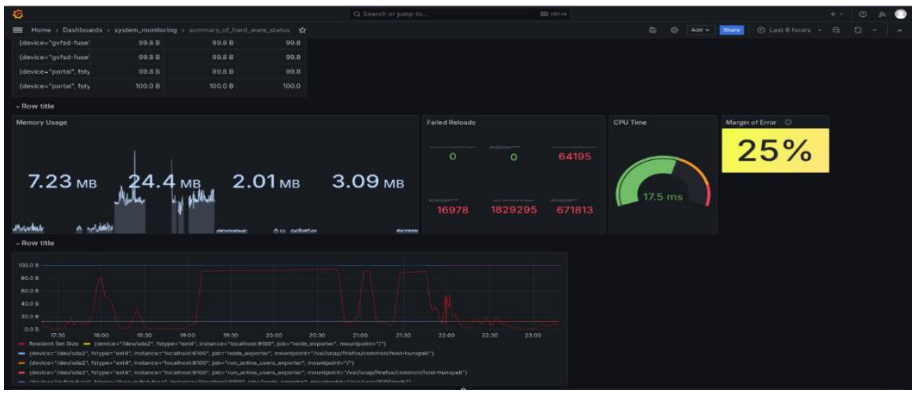
خروجی نهایی داشبورد:



شکل ۳۵-۴: تصویر کامل داشبورد گرافیکی خلاصه وضعیت سخت افزار (Grafana).



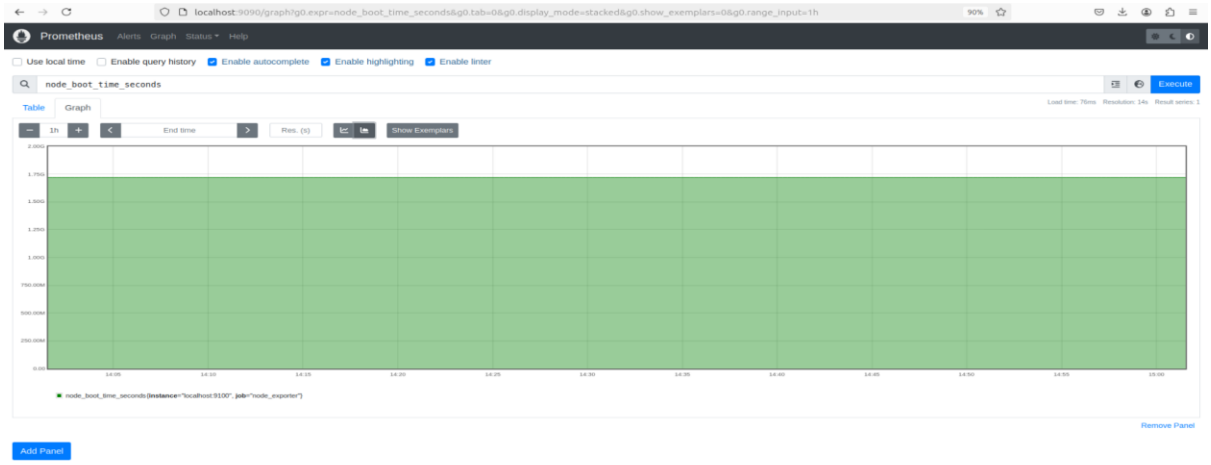
شکل ۳۶-۴: تصویر کامل داشبورد گرافیکی خلاصه وضعیت سخت افزار (Grafana).



شکل ۳۷-۴: تصویر کامل داشبورد گرافیکی خلاصه وضعیت سخت افزار (Grafana).

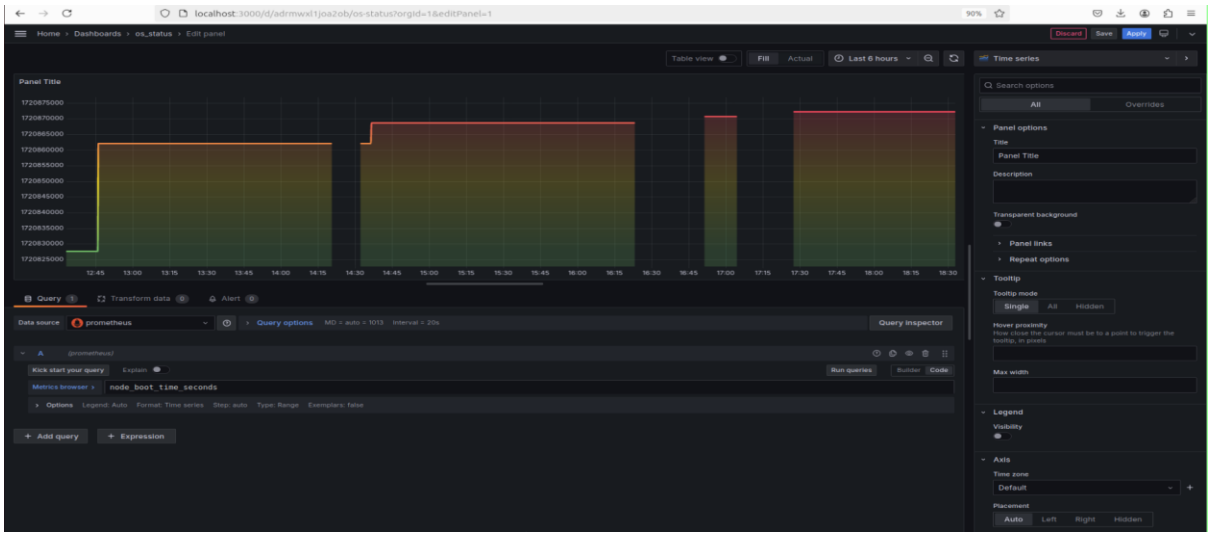
۲-۲-۴- خلاصه وضعیت سیستم عامل:

خروجی در Prometheus:



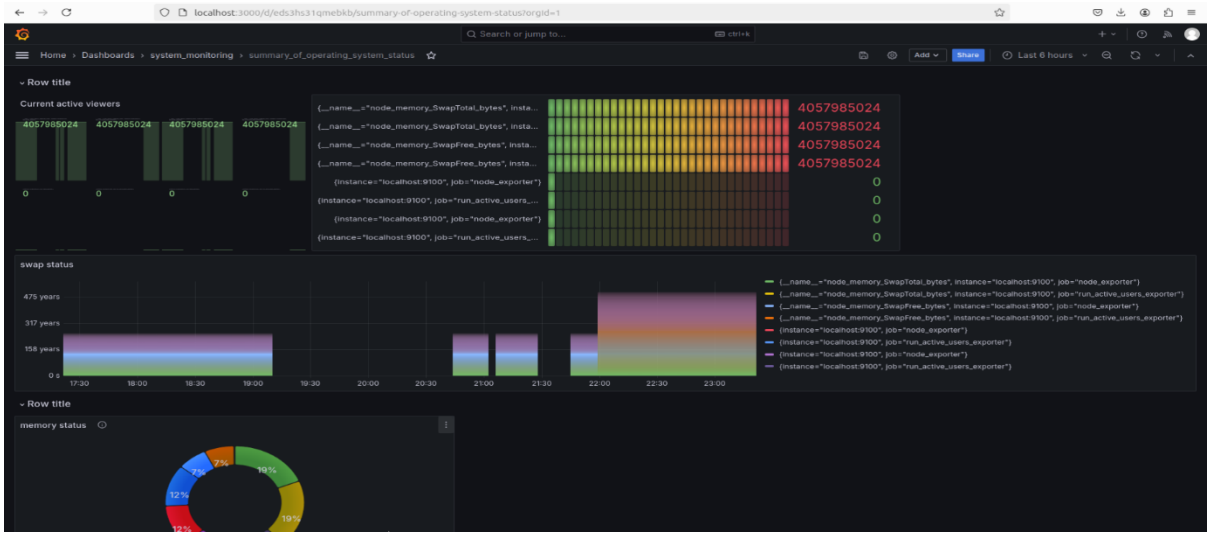
شکل ۳۸-۴: خلاصه وضعیت سیستم عامل (Prometheus).

خروجی در Grafana:

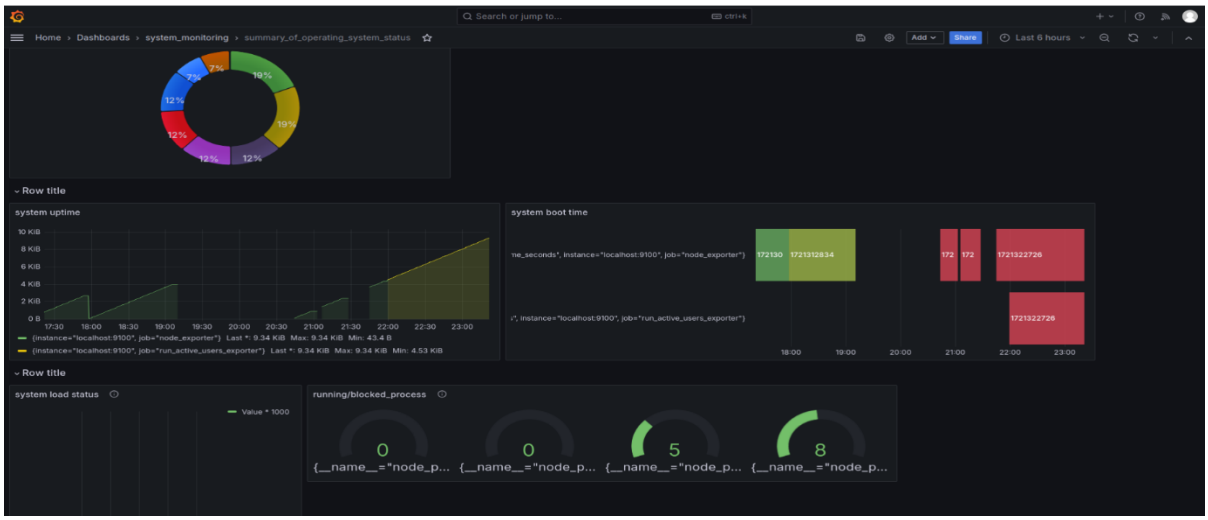


شکل ۳۹-۴: خلاصه وضعیت سیستم عامل (Grafana).

خروجی نهایی داشبورد:



شکل ۴۰-۴: تصویر کامل داشبورد گرافیکی خلاصه وضعیت سیستم عامل (Grafana).

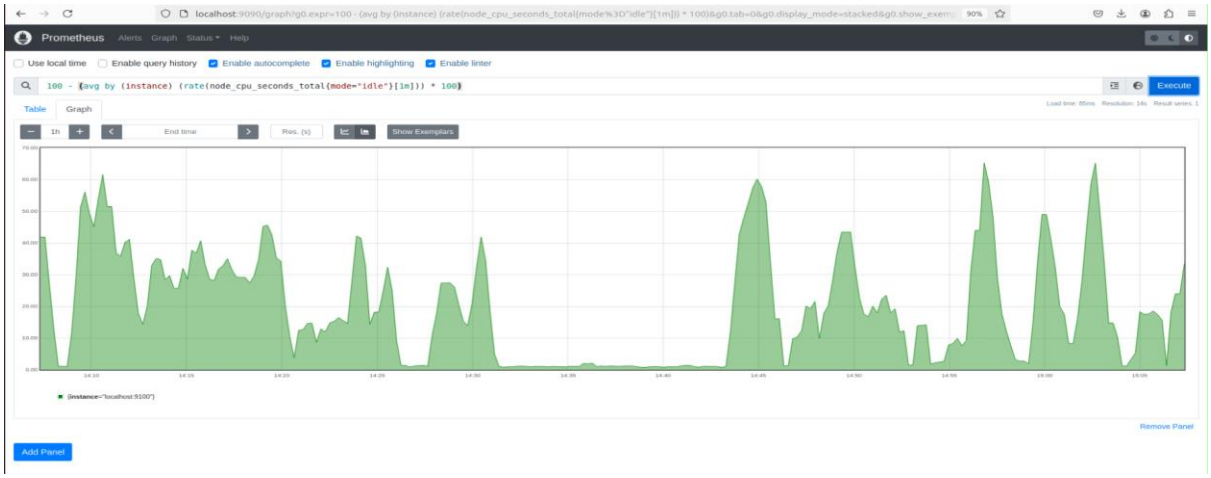


شکل ۴۱-۴: تصویر کامل داشبورد گرافیکی خلاصه وضعیت سیستم عامل (Grafana).

۳-۲-۴- میزان استفاده از پردازنده:

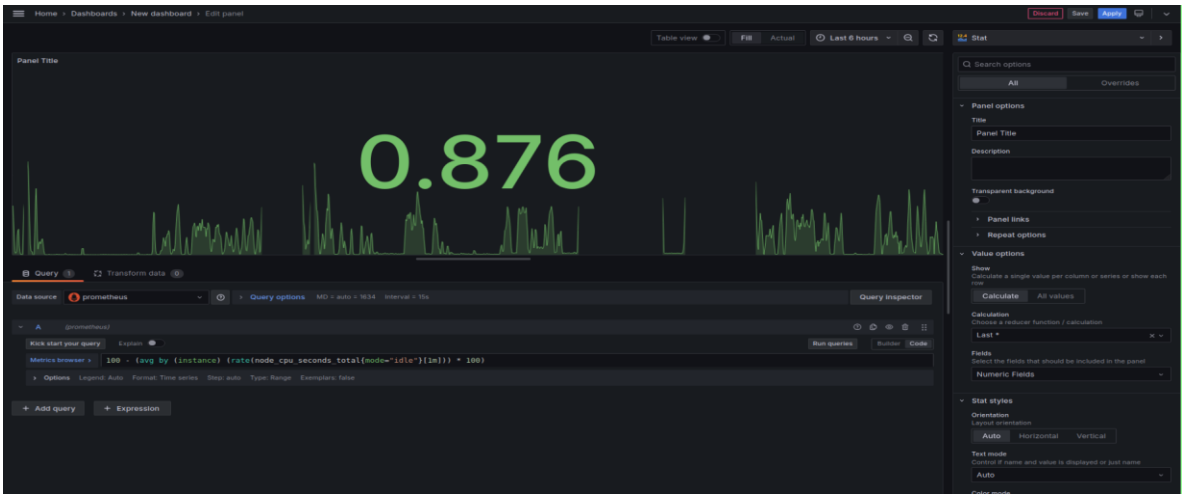
لحظه ای:

خروجی در Prometheus :



شکل ۴۲-۴: میزان استفاده از پردازنده لحظه ای (Prometheus).

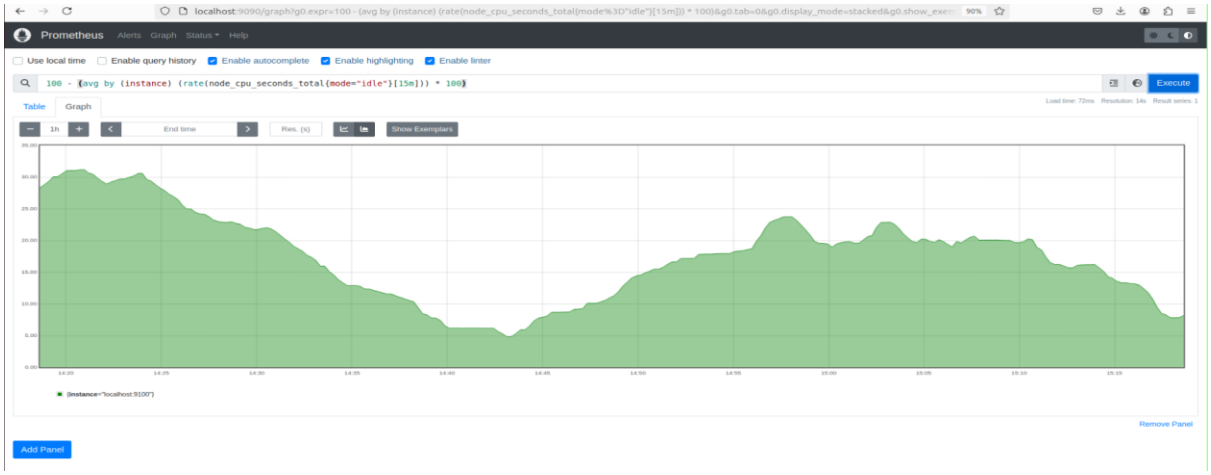
خروجی در Grafana :



شکل ۴۳-۴: میزان استفاده از پردازنده لحظه ای (Grafana).

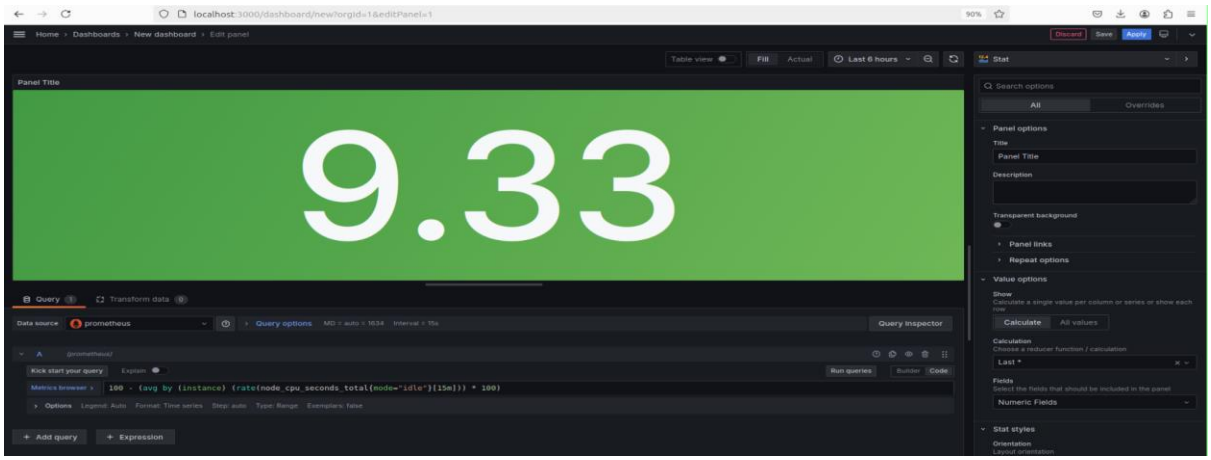
۱۵ دقیقه:

خروجی در Prometheus :



شکل ۴۴-۴: میزان استفاده از پردازنده ۱۵دقیقه (Prometheus).

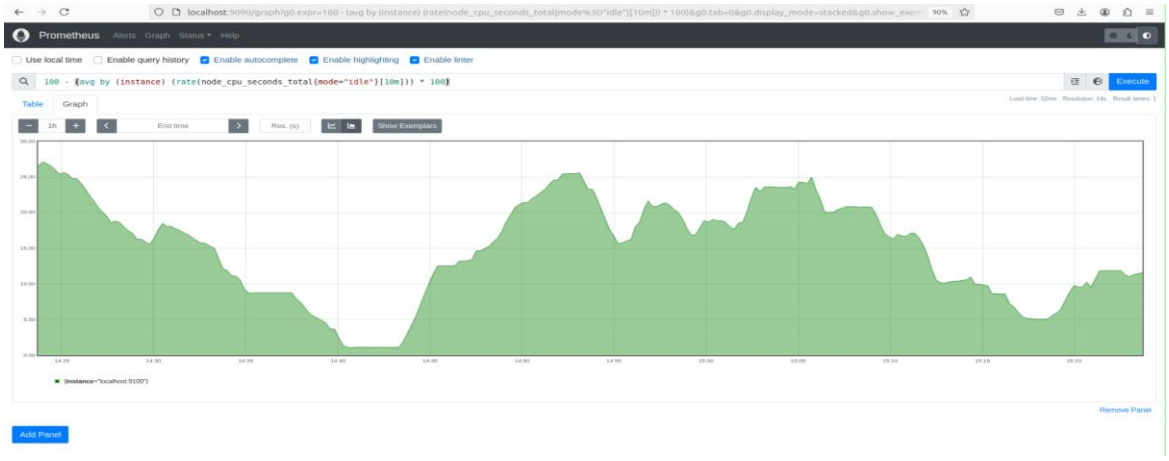
خروجی در Grafana :



شکل ۴۵-۴: میزان استفاده از پردازنده ۱۵دقیقه (Grafana).

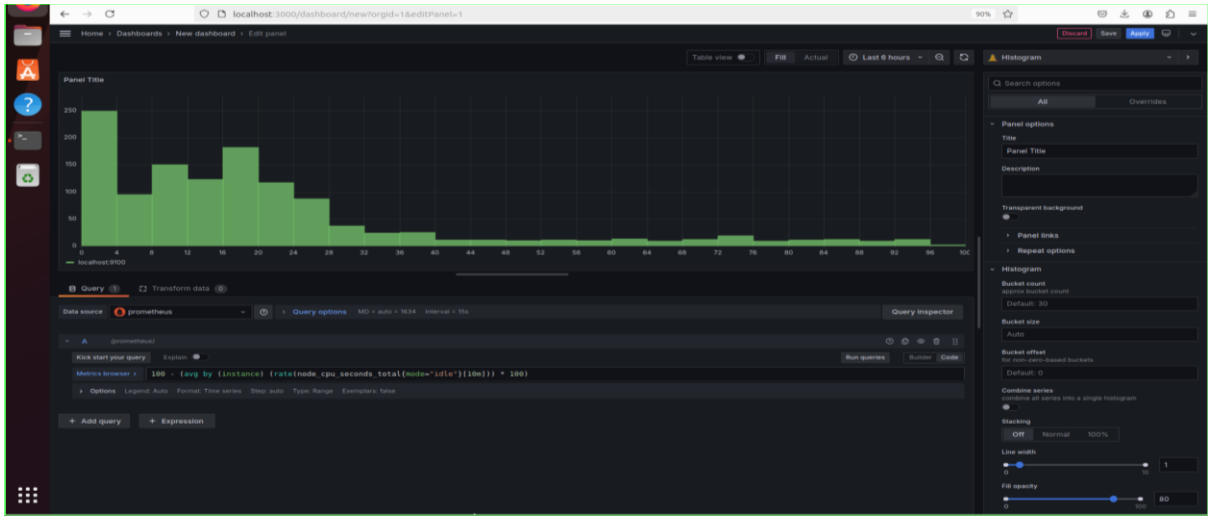
۱۰ دقیقه:

خروجی در Prometheus :



شکل ۴۶-۴: میزان استفاده از پردازنده ۱۰ دقیقه (Prometheus).

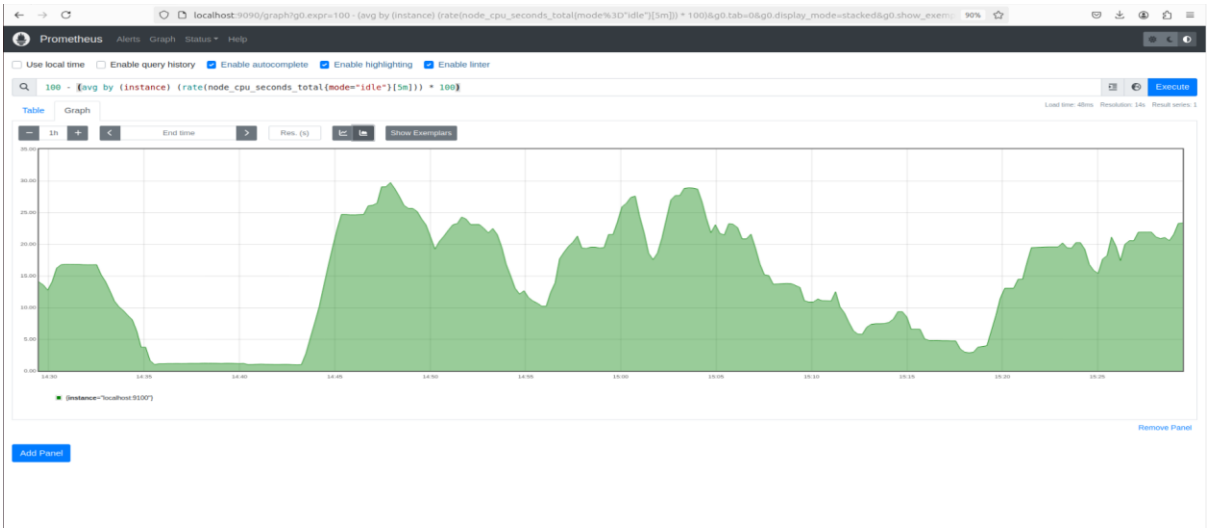
خروجی در Grafana :



شکل ۴۷-۴: میزان استفاده از پردازنده ۱۰دقیقه (Grafana).

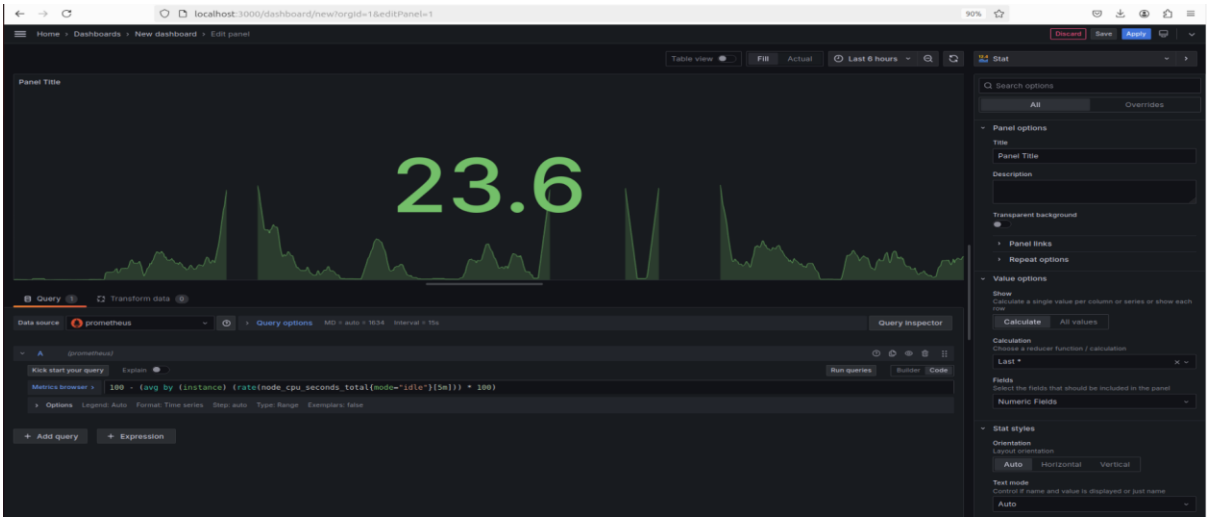
۵ دقیقه :

خروجی در Prometheus :



شکل ۴-۴۸: میزان استفاده از پردازنده ۵ دقیقه (Prometheus).

خروجی در Grafana :

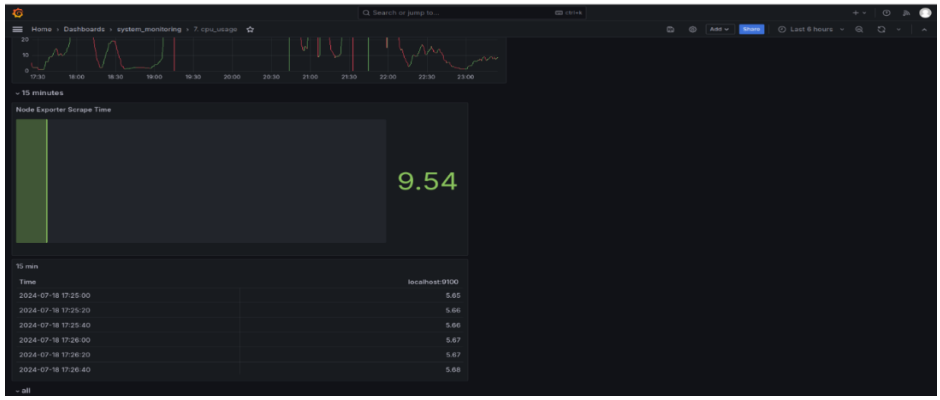


شکل ۴-۴۹: میزان استفاده از پردازنده ۵ دقیقه (Grafana).

خروجی نهایی داشبورد:

شکل ۵۰-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از پردازنده (Grafana).

شکل ۵۱-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از پردازنده (Grafana).

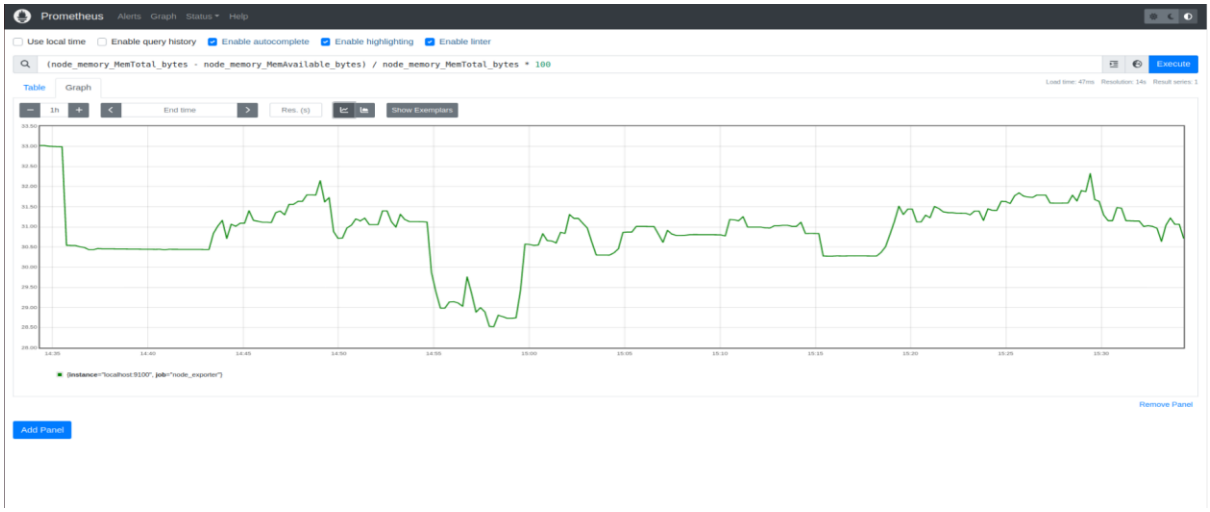


شکل ۵۲-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از پردازنده (Grafana).

۴-۲-۴- میزان استفاده از رم:

لحظه ای:

خروجی در Prometheus:



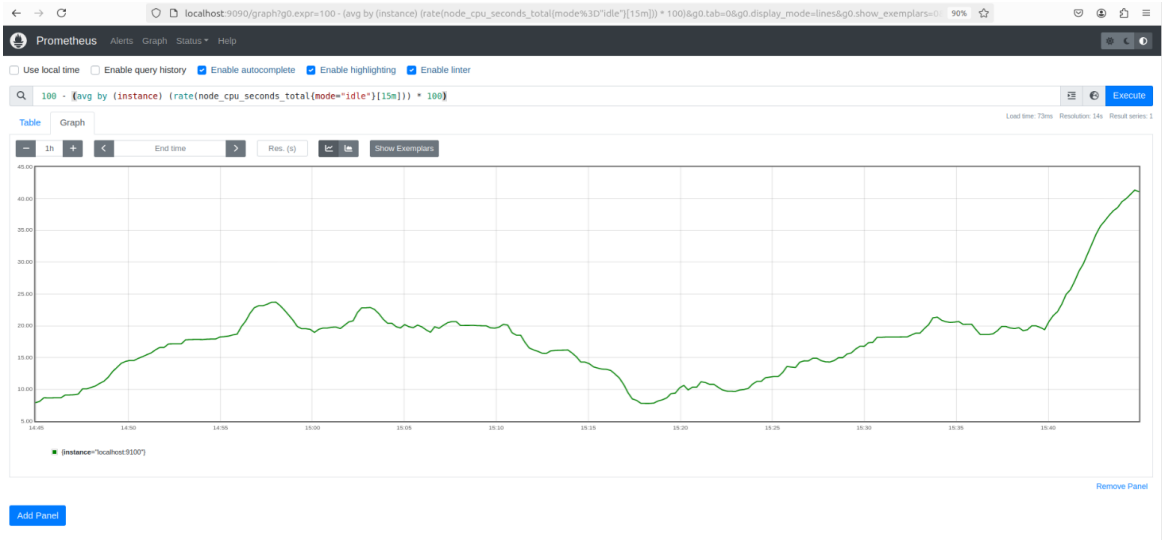
شکل ۵۳-۴: میزان استفاده از رم لحظه ای (Prometheus).

خروجی در Grafana:

شکل ۵۴-۴: میزان استفاده از رم لحظه ای (Grafana).

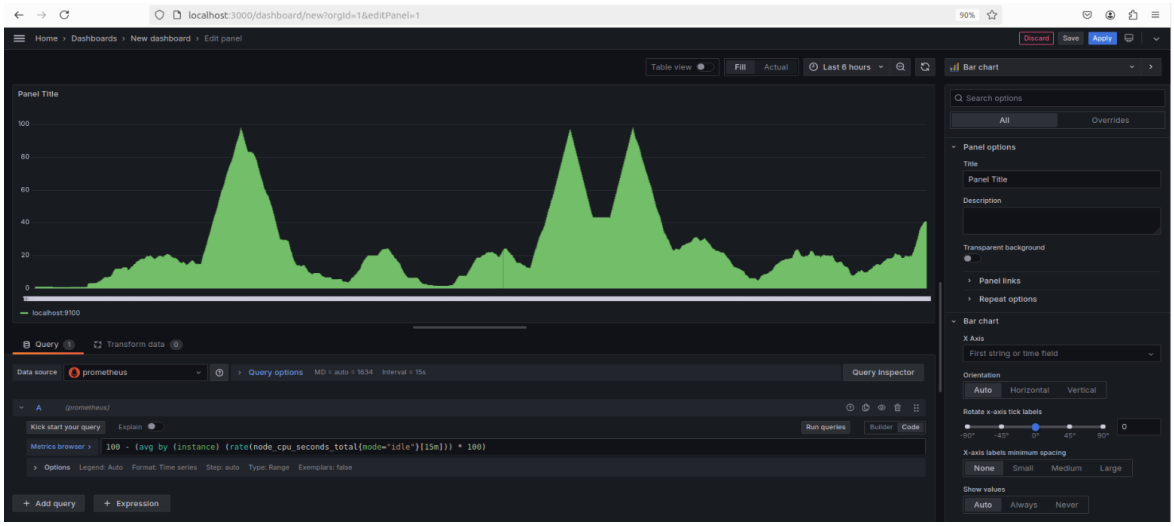
۱۵ دقیقه:

خروجی در Prometheus :



شکل ۵۵-۴: میزان استفاده از رم ۱۵ دقیقه (Prometheus).

خروجی در Grafana :



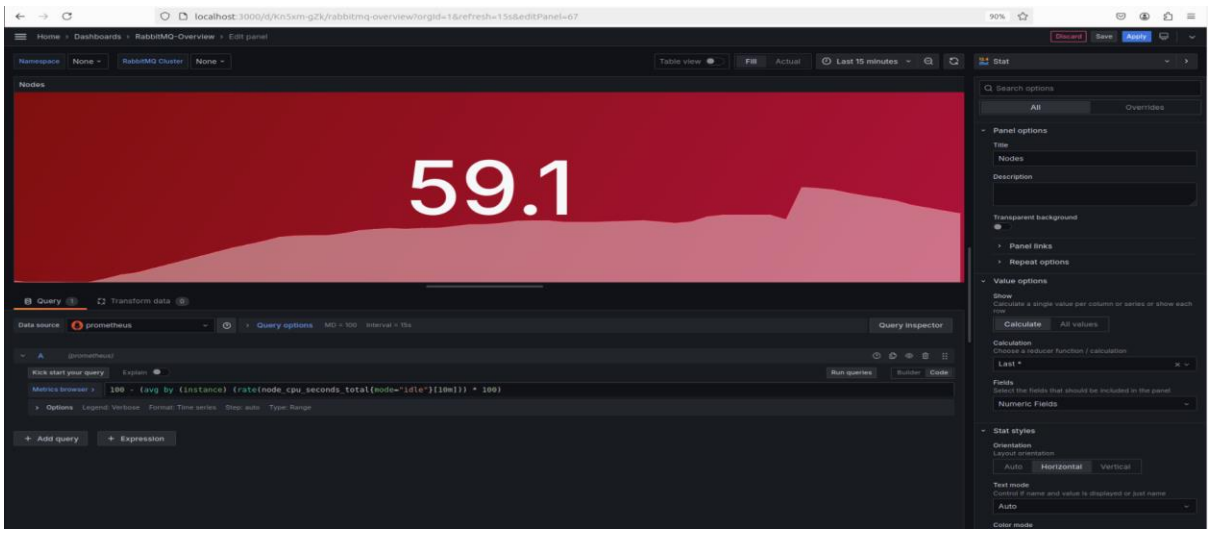
شکل ۵۶-۴: میزان استفاده از رم ۱۵ دقیقه (Grafana).

۱۰ دقیقه:

خروجی در Prometheus:

شکل ۵۷-۴: میزان استفاده از رم ۱۰ دقیقه (Prometheus).

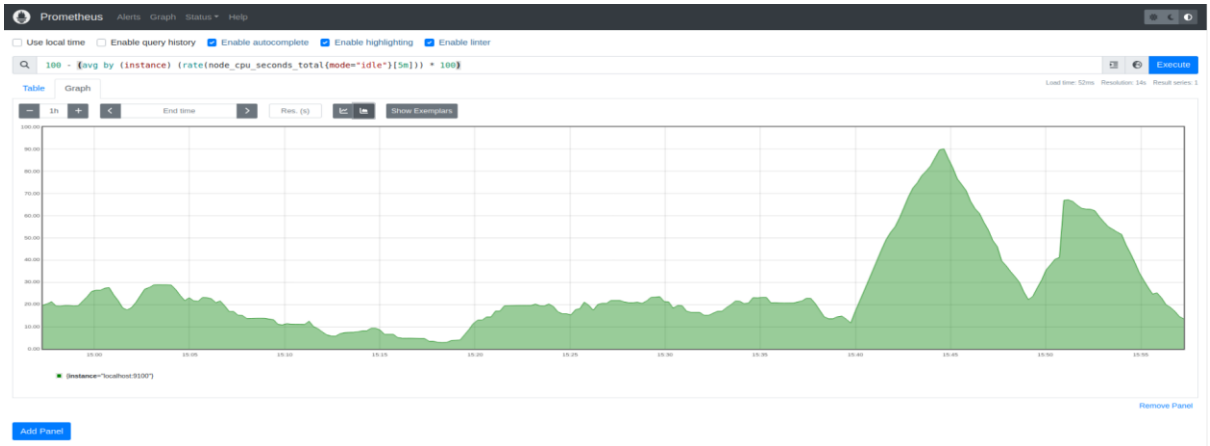
خروجی در Grafana:



شکل ۵۸-۴: میزان استفاده از رم ۱۰ دقیقه (Grafana).

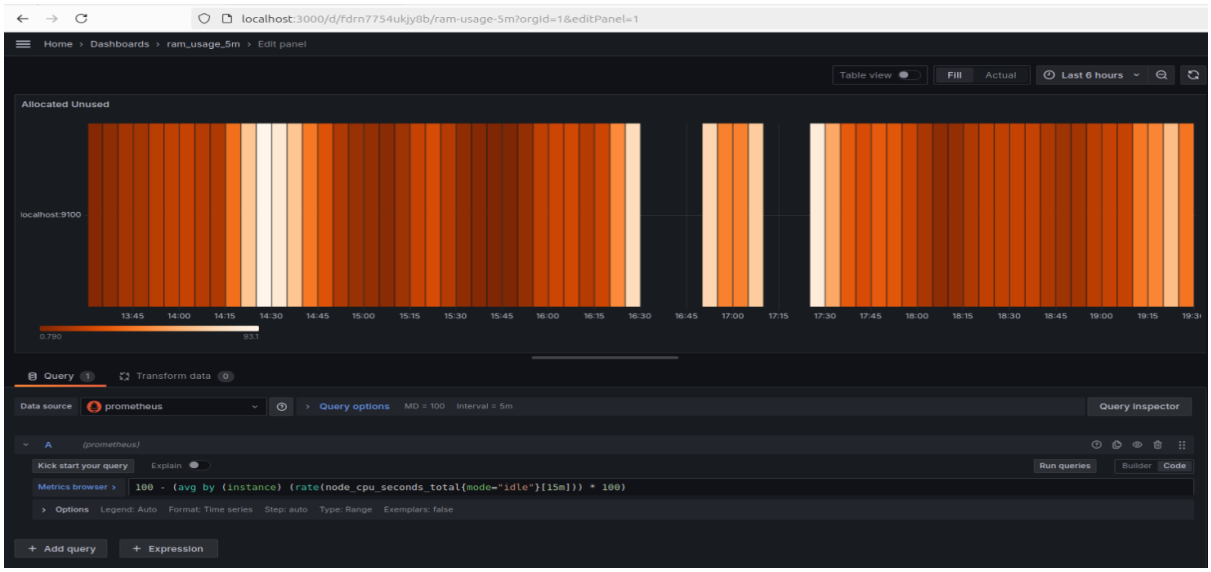
۵ دقیقه:

خروجی در Prometheus :



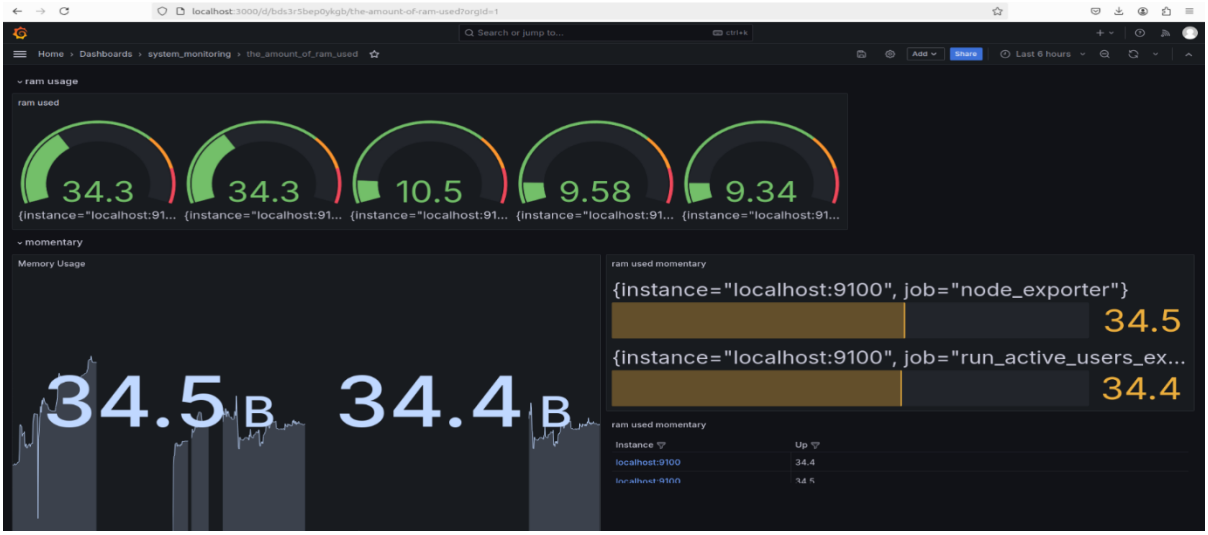
شکل ۵۹-۴: میزان استفاده از رم ۵ دقیقه (Prometheus).

خروجی در Grafana :



شکل ۶۰-۴: میزان استفاده از رم ۵ دقیقه (Grafana).

خروجی نهایی داشبورد:



شکل ۶۱-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از رم (Grafana).

شکل ۶۲-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از رم (Grafana).

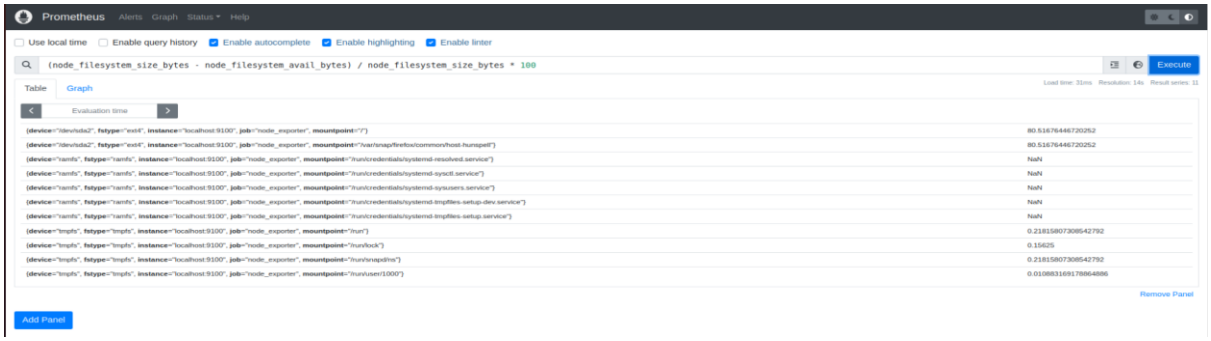
شکل ۶۳-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از رم (Grafana).

شکل ۶۴-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از رم (Grafana).

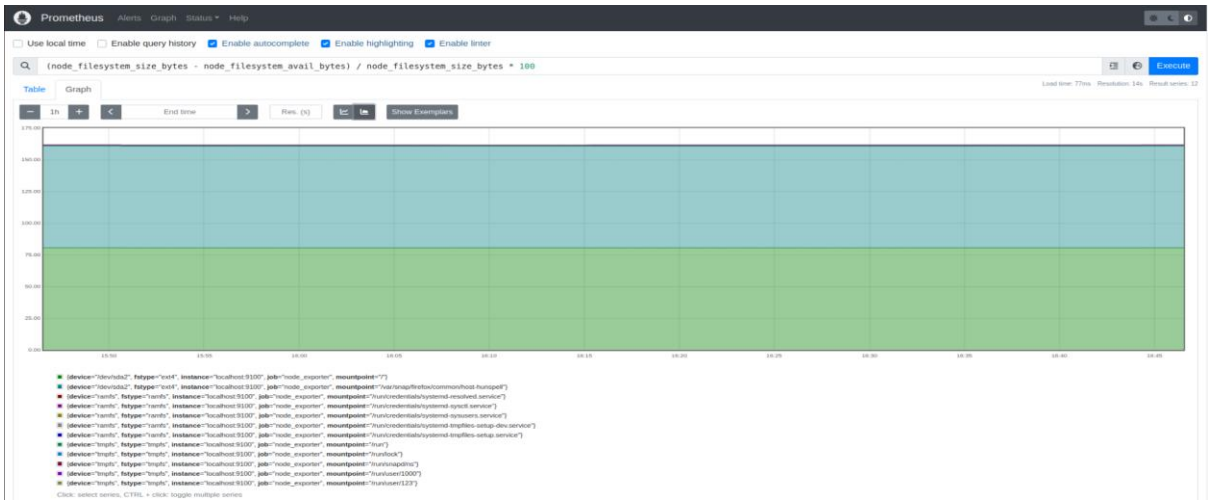
۵-۲-۴- میزان استفاده از هارد:

لحظه ای:

خروجی در Prometheus :

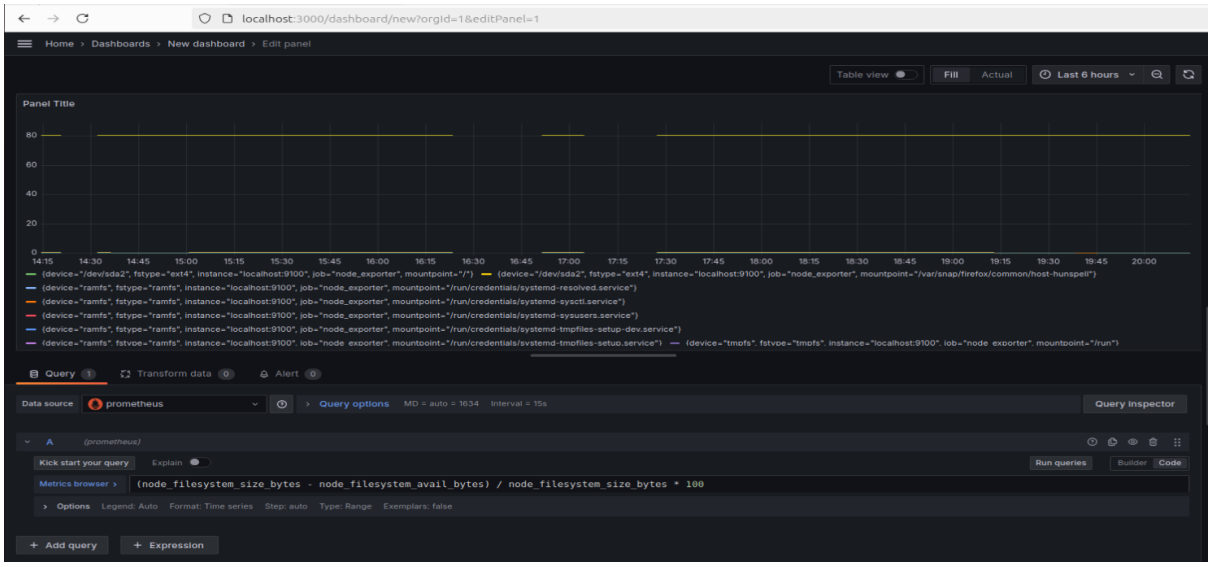


شکل ۶۵-۴: میزان استفاده از هارد لحظه ای (Prometheus).

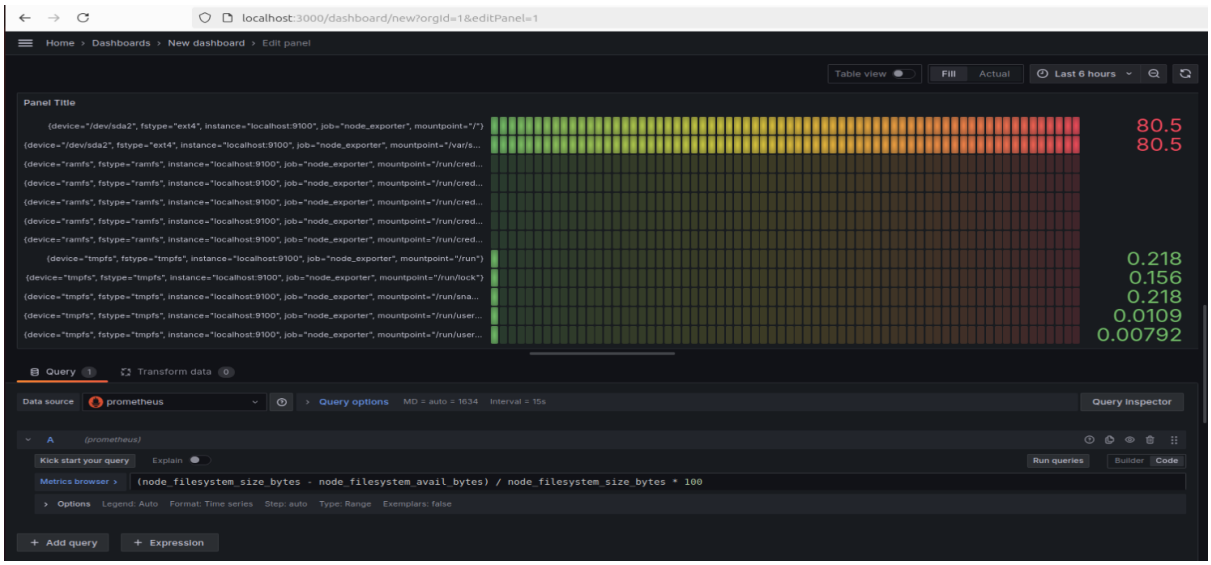


شکل ۶۶-۴: میزان استفاده از هارد لحظه ای (Prometheus).

خروجی در Grafana :



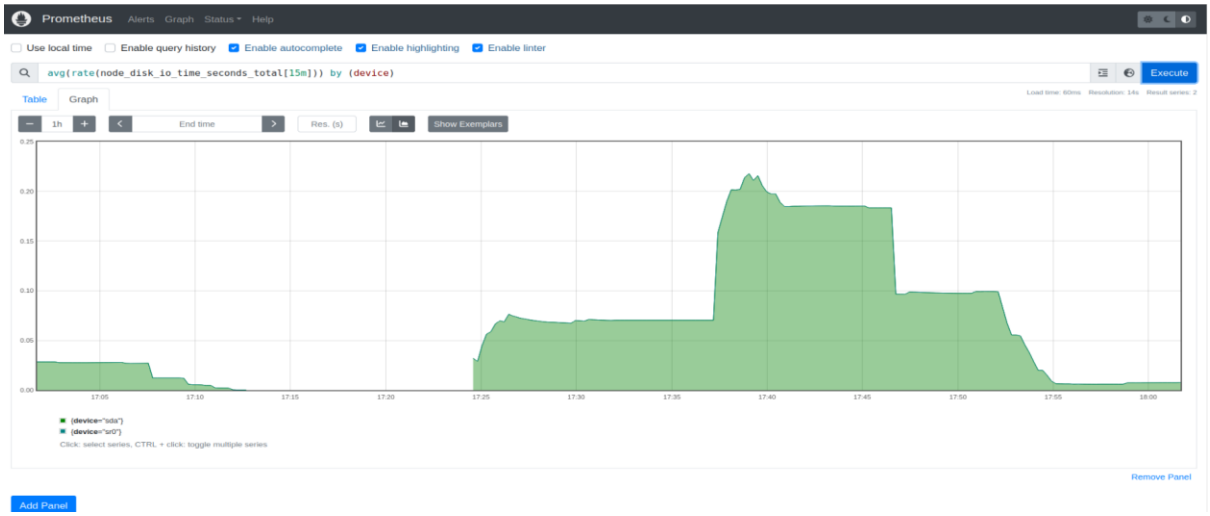
شکل ۶۷-۴: میزان استفاده از هارد لحظه ای (Grafana).



شکل ۶۸-۴: میزان استفاده از هارد لحظه ای (Grafana).

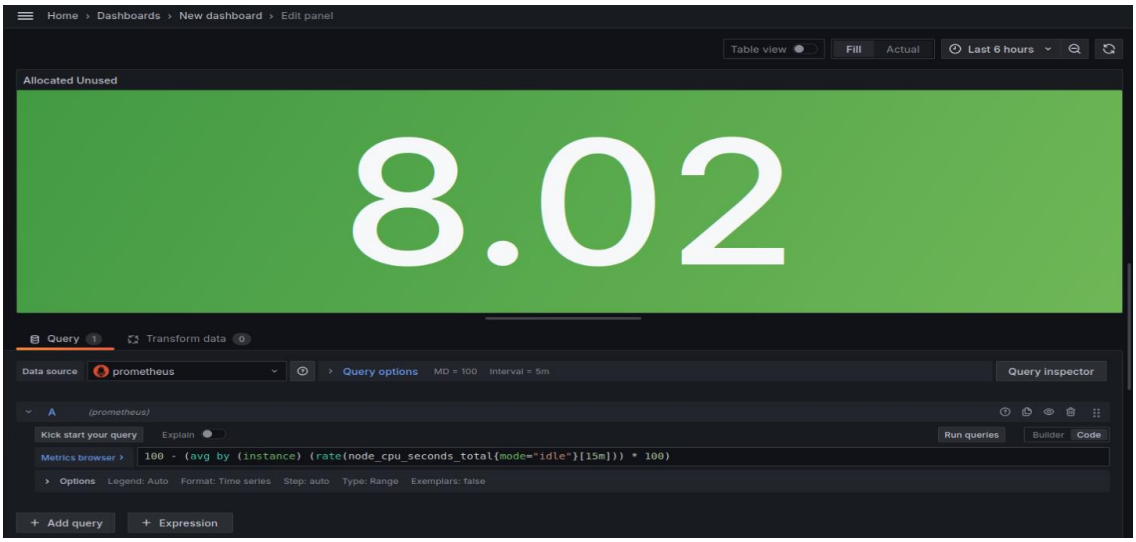
۱۵ دقیقه:

خروجی در Prometheus:



شکل ۶۹-۴: میزان استفاده از هارد ۱۵ دقیقه (Prometheus).

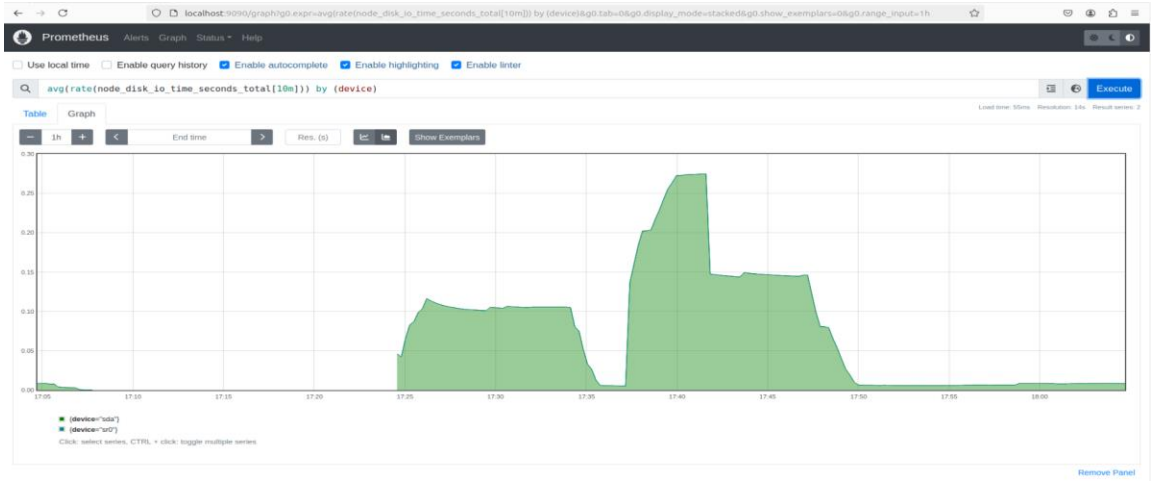
خروجی در Grafana:



شکل ۷۰-۴: میزان استفاده از هارد ۱۵ دقیقه (Grafana).

۱۰ دقیقه:

خروجی در Prometheus:



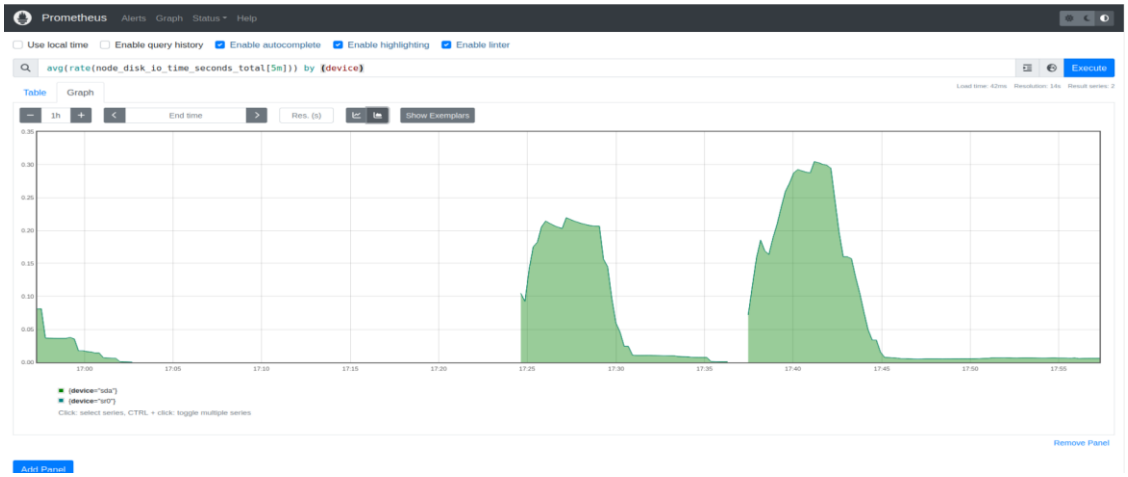
شکل ۷۱-۴: میزان استفاده از هارد ۱۰ دقیقه (Prometheus).

خروجی در Grafana:

شکل ۷۲-۴: میزان استفاده از هارد ۱۰ دقیقه (Grafana).

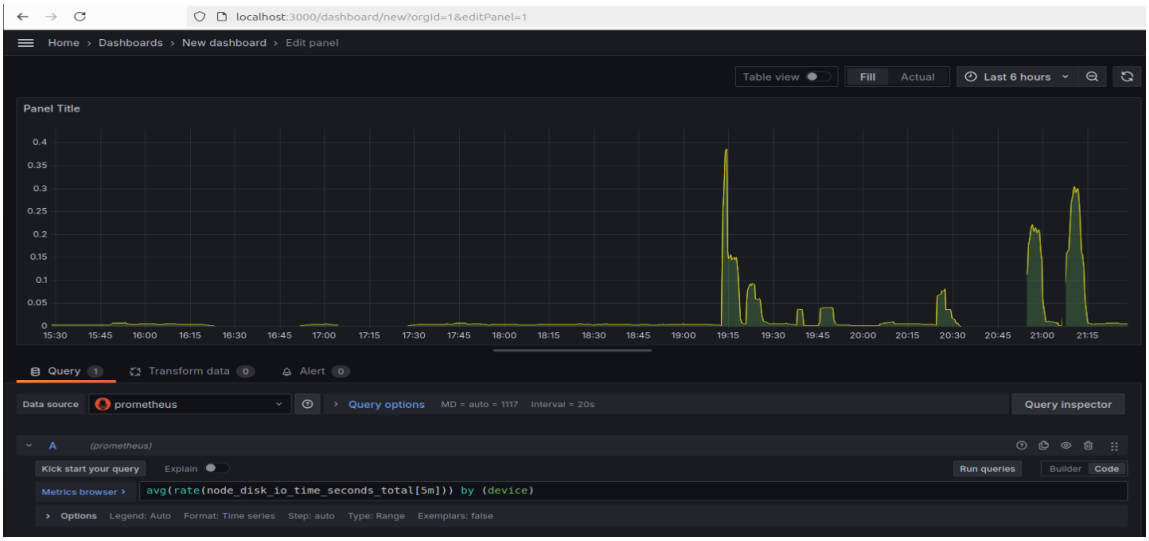
۵ دقیقه:

خروجی در Prometheus :



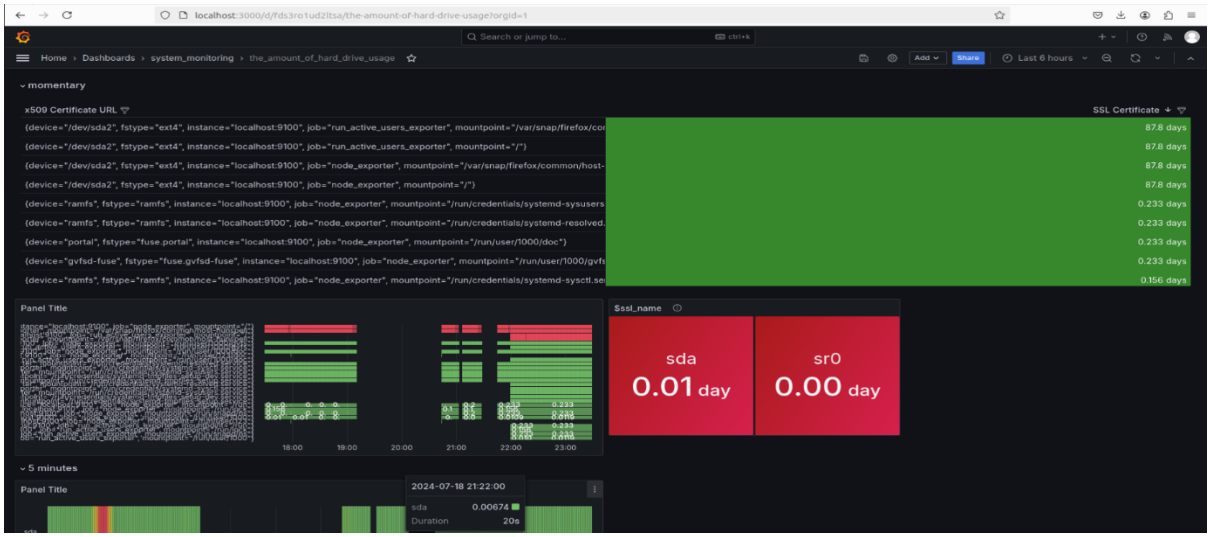
شکل ۷۳-۴: میزان استفاده از هارد ۵ دقیقه (Prometheus).

خروجی در Grafana :

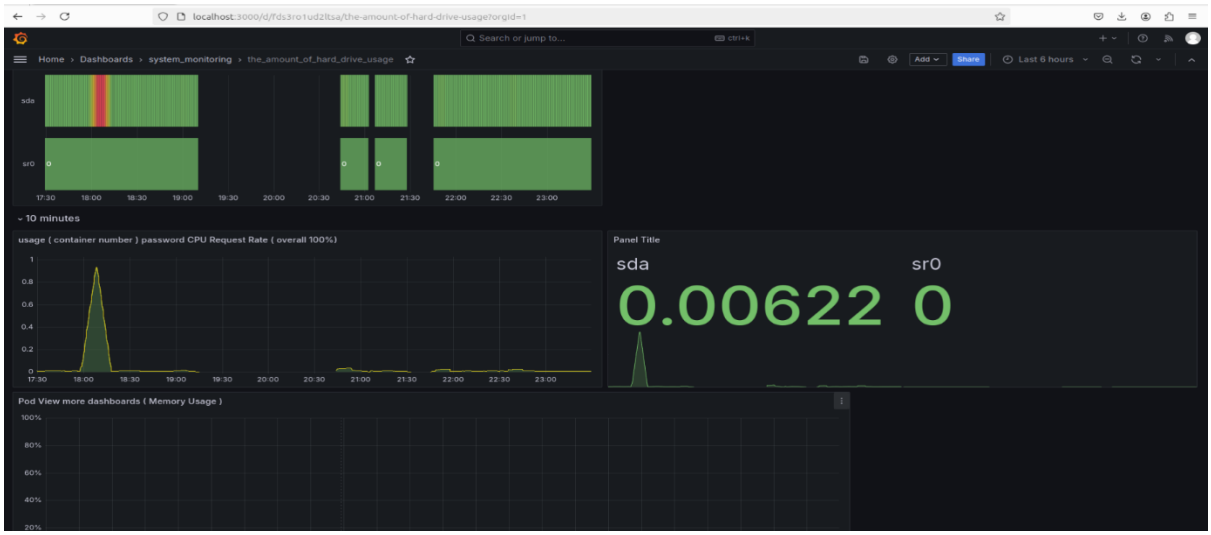


شکل ۷۴-۴: میزان استفاده از هارد ۵ دقیقه (Grafana).

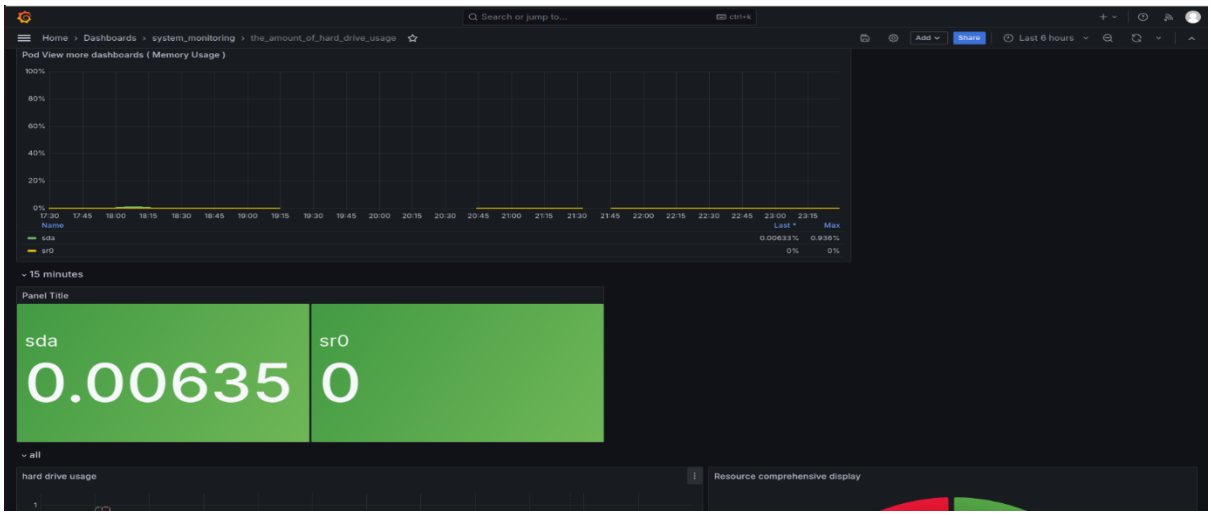
خروجی نهایی داشبورد:



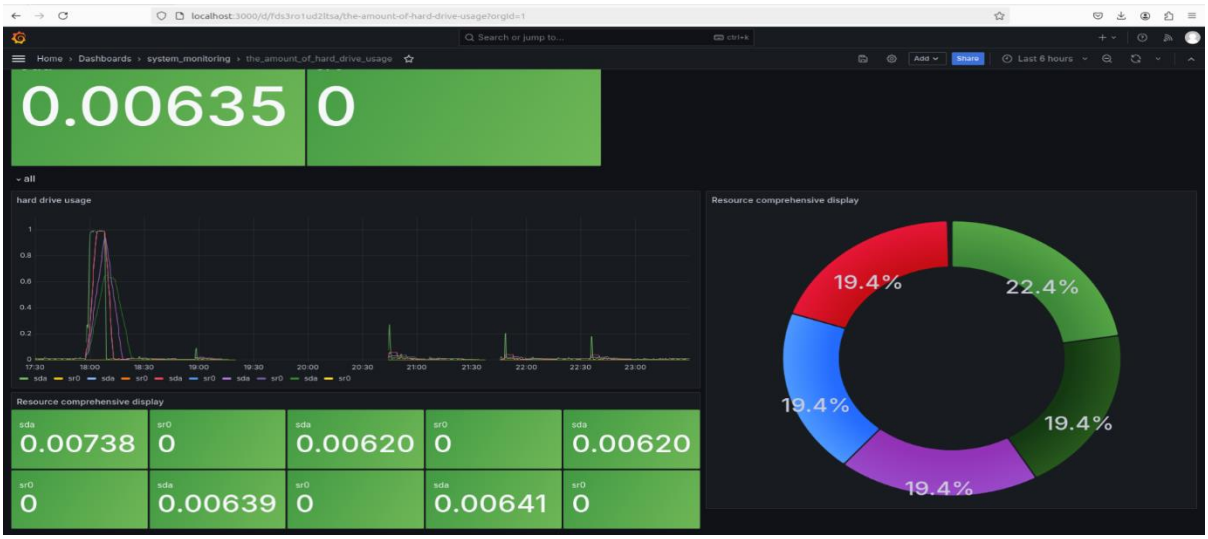
شکل ۷۵-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از هارد (Grafana).



شکل ۷۶-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از هارد (Grafana).



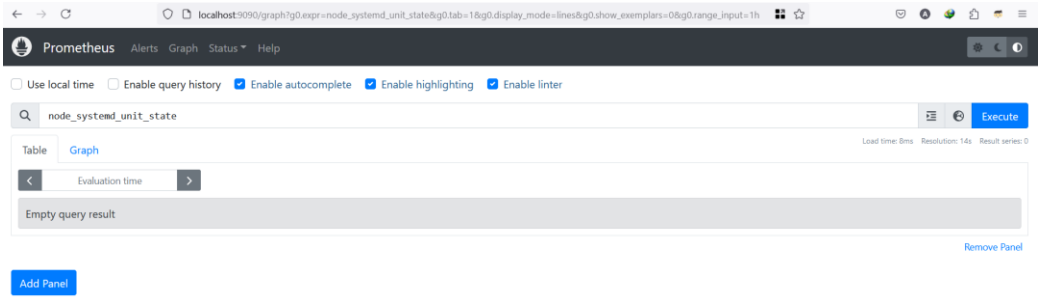
شکل ۷۷-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از هارد (Grafana).



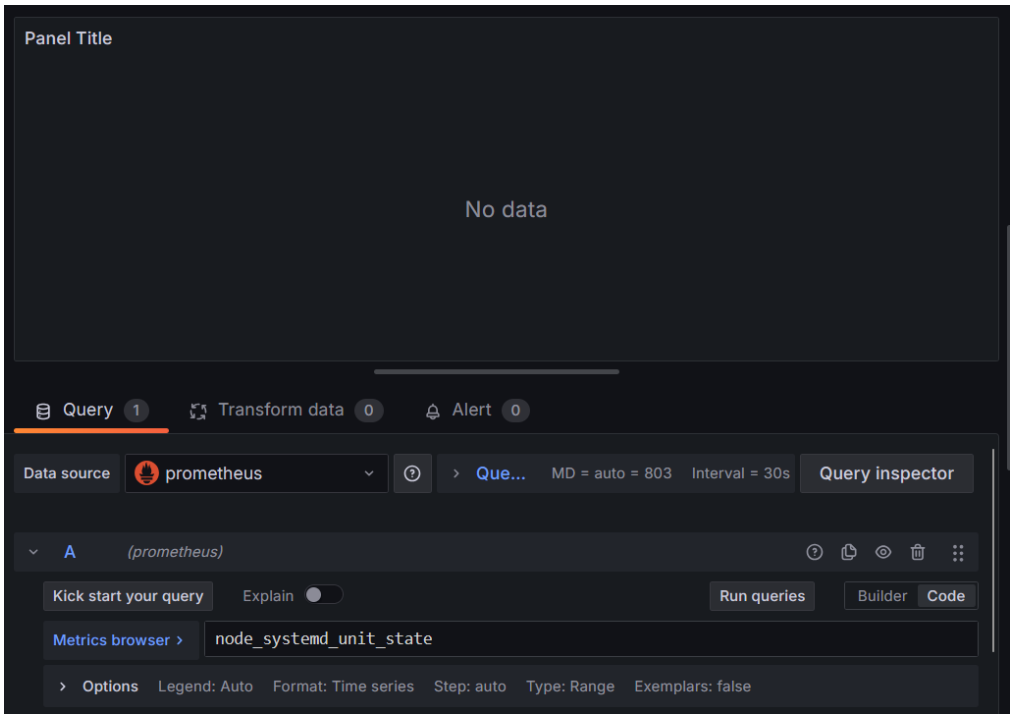
شکل ۷۸-۴: تصویر کامل داشبورد گرافیکی میزان استفاده از هارد (Grafana).

۶-۲-۴- تنظیم دسترسی های خاص

برخی از دستورات مانند میزان استفاده از هارد لحظه ای (تاریخچه ۵،۱۰،۱۵ دقیقه) و لیست کاربران سیستم (کاربران فعال) و لیست سخت افزارهای متصل و لیست سرویسهای در حال اجرا و لاگهای سطح سیستم عامل با دسترسی پرومتئوس قابل اجرا نیستند.



شکل ۷۹-۴: عدم دسترسی به محتوا در(Prometheus).



شکل ۸۰-۴: عدم دسترسی به محتوا در(Grafana).

برای اجرای این دستورات نیاز هست از کامند های لینوکسی بهره ببریم که در ادامه بطور کامل توضیح داده می شود:

ابتدا با استفاده از دستور nano یک فایل ایجاد و دستورات مربوط به موارد خواسته شده را با استفاده از کامند های لینوکسی در آن قرار می دهیم:

nano run_active_users_exporter.sh

شکل ۸۱-۴: وارد کردن دستورات در nano.

حال با استفاده از دستورات لینوکسی زیر دسترسی های لازم را برای فایلی که ایجاد کردیم تعیین می کنیم:

nano run_active_users_exporter.sh

chmod +x run_active_users_exporter.sh

سپس فایل ساخته شده را اجرا می کنیم تا از صحت دستورات مطمئن شویم.

./run_active_users_exporter.sh

خروجی فایل در صورت صحت دستورات و اجرا به شکل زیر خواهد بود:

```
amir@amir:~$ ./run_active_users_exporter.sh
Active System Users:
amir

Connected Hardware:
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 001 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

شکل ۸۲-۴: صحت اجرای دستورات.

```
Running Services:
  UNIT                          LOAD   ACTIVE SUB     DESCRIPTION
  accounts-daemon.service       loaded active running Accounts Service
  avahi-daemon.service          loaded active running Avahi mDNS/DNS-SD Stack
  blackbox_exporter.service     loaded active running Blackbox Exporter
  colord.service                loaded active running Manage, Install and Generate Color Profiles
  containerd.service            loaded active running containerd container runtime
  cron.service                  loaded active running Regular background program processing daemon
  cups-browsed.service          loaded active running Make remote CUPS printers available locally
  cups.service                  loaded active running CUPS Scheduler
  dbus.service                  loaded active running D-Bus System Message Bus
  docker.service                loaded active running Docker Application Container Engine
  fwupd.service                 loaded active running Firmware update daemon
  gdm.service                   loaded active running GNOME Display Manager
  grafana-server.service        loaded active running Grafana instance
  influxdb.service              loaded active running InfluxDB is an open-source, distributed, time series database
  irqbalance.service            loaded active running irqbalance daemon
  kerneloops.service            loaded active running Tool to automatically collect and submit kernel crash signatures
  loki.service                  loaded active running Loki service
  ModemManager.service          loaded active running Modem Manager
  networkd-dispatcher.service   loaded active running Dispatcher daemon for systemd-networkd
  NetworkManager.service        loaded active running Network Manager
  nginx.service                 loaded active running A high performance web server and a reverse proxy server
  node_exporter.service         loaded active running Node Exporter
  polkit.service                loaded active running Authorization Manager
  postfix@-.service             loaded active running Postfix Mail Transport Agent (instance -)
  power-profiles-daemon.service loaded active running Power Profiles daemon
  prometheus.service            loaded active running Prometheus
  promtail.service              loaded active running Promtail service
  rsyslog.service               loaded active running System Logging Service
  rtkit-daemon.service          loaded active running RealtimeKit Scheduling Policy Service
  snapd.service                 loaded active running Snap Daemon
  switcheroo-control.service    loaded active running Switcheroo Control Proxy service
  systemd-journald.service      loaded active running Journal Service
  systemd-logind.service        loaded active running User Login Management
  systemd-oomd.service          loaded active running Userspace Out-Of-Memory (OOM) Killer
  systemd-resolved.service      loaded active running Network Name Resolution
  systemd-timesyncd.service     loaded active running Network Time Synchronization
  systemd-udevd.service         loaded active running Rule-based Manager for Device Events and Files
  telegraf.service              loaded active running The plugin-driven server agent for reporting metrics into InfluxDB
  tftpd-hpa.service             loaded active running LSB: HPA's tftp server
  udisks2.service               loaded active running Disk Manager
  unattended-upgrades.service   loaded active running Unattended Upgrades Shutdown
  upower.service                loaded active running Daemon for power management
  user@1000.service             loaded active running User Manager for UID 1000
  wpa_supplicant.service        loaded active running WPA supplicant
```

شکل ۸۳-۴: صحت اجرای دستورات.

```
LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
44 loaded units listed.
```

شکل ۸۴-۴: صحت اجرای دستورات.

```
Operating System Logs:
Jul 17 23:43:34 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:43:34 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:43:44 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:43:44 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:43:54 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:43:54 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:00 amir telegraf[1118]: 2024-07-17T20:14:00Z E! [inputs.ping] Error in plugin: host http://google.com: /usr/bin/ping: http://google.com: Name or service not known, exit
Jul 17 23:44:00 amir telegraf[1118]: 2024-07-17T20:14:00Z E! [inputs.ping] Error in plugin: host http://digikala.com: /usr/bin/ping: http://digikala.com: Name or service not known, e
Jul 17 23:44:04 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:04 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:04 amir loki[1169]: level=info ts=2024-07-17T20:14:04.7514900 8Z caller=table_manager.go:136 index-store=tsdb-2020-10-24 msg="uploading tables"
Jul 17 23:44:14 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:14 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:24 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:24 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:34 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:34 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:44 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:44 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:44:54 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:44:54 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:00 amir telegraf[1118]: 2024-07-17T20:15:00Z E! [inputs.ping] Error in plugin: host http://google.com: /usr/bin/ping: http://google.com: Name or service not known, exit
Jul 17 23:45:00 amir telegraf[1118]: 2024-07-17T20:15:00Z E! [inputs.ping] Error in plugin: host http://digikala.com: /usr/bin/ping: http://digikala.com: Name or service not known, e
Jul 17 23:45:04 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:04 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:04 amir loki[1169]: level=info ts=2024-07-17T20:15:04.751882582Z caller=table_manager.go:136 index-store=tsdb-2020-10-24 msg="uploading tables"
Jul 17 23:45:06 amir systemd-timesyncd[855]: Timed out waiting for reply from 185.125.190.57:123 (ntp.ubuntu.com).
Jul 17 23:45:14 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:14 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:16 amir systemd-timesyncd[855]: Timed out waiting for reply from 185.125.190.56:123 (ntp.ubuntu.com).
Jul 17 23:45:24 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:24 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:27 amir systemd-timesyncd[855]: Timed out waiting for reply from 185.125.190.58:123 (ntp.ubuntu.com).
Jul 17 23:45:34 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:34 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:37 amir systemd-timesyncd[855]: Timed out waiting for reply from 91.189.91.157:123 (ntp.ubuntu.com).
Jul 17 23:45:44 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:44 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:45:54 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:45:54 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:00 amir telegraf[1118]: 2024-07-17T20:16:00Z E! [inputs.ping] Error in plugin: host http://google.com: /usr/bin/ping: http://google.com: Name or service not known, exit
Jul 17 23:46:00 amir telegraf[1118]: 2024-07-17T20:16:00Z E! [inputs.ping] Error in plugin: host http://digikala.com: /usr/bin/ping: http://digikala.com: Name or service not known, e
Jul 17 23:46:04 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:04 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:04 amir loki[1169]: level=info ts=2024-07-17T20:16:04.751955077Z caller=table_manager.go:136 index-store=tsdb-2020-10-24 msg="uploading tables"
Jul 17 23:46:14 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:14 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:24 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:24 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:34 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:34 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:44 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:44 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:46:54 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:46:54 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:47:00 amir telegraf[1118]: 2024-07-17T20:17:00Z E! [inputs.ping] Error in plugin: host http://google.com: /usr/bin/ping: http://google.com: Name or service not known, exit
Jul 17 23:47:00 amir telegraf[1118]: 2024-07-17T20:17:00Z E! [inputs.ping] Error in plugin: host http://digikala.com: /usr/bin/ping: http://digikala.com: Name or service not known, e
Jul 17 23:47:04 amir influxd[1115]: [httpd] 127.0.0.1 - - [17/Jul/2024:23:47:04 +0330] "POST /write?consistency=any&db=ping_data HTTP/1.1" 204 0 "-" "Telegraf/1.22.3+ds1-0ubuntu2 Go/
Jul 17 23:47:04 amir loki[1169]: level=info ts=2024-07-17T20:17:04.751165367Z caller=table_manager.go:232 index-store=tsdb-2020-10-24 msg="syncing tables"
Jul 17 23:47:04 amir loki[1169]: level=info ts=2024-07-17T20:17:04.751199681Z caller=table_manager.go:280 index-store=tsdb-2020-10-24 msg="query readiness setup completed" duration=2
Jul 17 23:47:04 amir loki[1169]: level=info ts=2024-07-17T20:17:04.7512350 3Z caller=table_manager.go:136 index-store=tsdb-2020-10-24 msg="uploading tables"
Jul 17 23:47:05 amir loki[1169]: level=info ts=2024-07-17T20:17:05.081050931Z caller=checkpoint.go:611 msg="starting checkpoint"
Jul 17 23:47:05 amir loki[1169]: level=info ts=2024-07-17T20:17:05.0821050 4Z caller=checkpoint.go:336 msg="attempting checkpoint for" dir=/tmp/loki/wal/checkpoint.000001
Jul 17 23:47:05 amir loki[1169]: level=info ts=2024-07-17T20:17:05.089217114Z caller=checkpoint.go:498 msg="atomic checkpoint finished" old=/tmp/loki/wal/checkpoint.000001.tmp new=/t
Jul 17 23:47:10 amir loki[1169]: level=info ts=2024-07-17T20:17:10.307338646Z caller=compactor.go:523 msg="compactor startup delay completed"
Jul 17 23:47:10 amir loki[1169]: ts=2024-07-17T20:17:10.3074367 09Z caller=spanlogger.go:109 level=info msg="building table names cache"
Jul 17 23:47:10 amir loki[1169]: ts=2024-07-17T20:17:10.30765527 caller=spanlogger.go:109 level=info msg="table names cache built" duration=71.928us
```

شکل ۸۵-۴: صحت اجرای دستورات.

حال داخل تنظیمات پرومتئوس وارد می شویم و یک جاب جدید در قسمت نود اکسپورتر تعریف و فایل خود و پورتی که میخواهیم به آن تخصیص دهیم را وارد می کنیم:

```
rule_files: []

scrape_configs:
  - job_name: 'prometheus'
    static_configs:
      - targets: ['localhost:9090']

  - job_name: 'node_exporter'
    static_configs:
      - targets: ['localhost:9100']
  - job_name: 'run_active_users_exporter'
    static_configs:
      - targets: ['localhost:9101']
```

شکل ۸۶-۴: اضافه کردن به تنظیمات نود اکسپورتر.

حال باید ابتدا دستورات نوشته شده را بر روی یک لوکال سرور بالا بیاوریم تا پرومتئوس بتواند از آنها استفاده کند:

بدین منظور برنامه netcat را نصب می کنیم تا یک سرور TCP ساده را بر روی آن پورت ۹۱۰۱ اجرا کنیم.

سپس یکی از نسخه های را نصب می کنیم:

یا برای نصب netcat-traditional:

```
sudo apt update
sudo apt install netcat-traditional
```

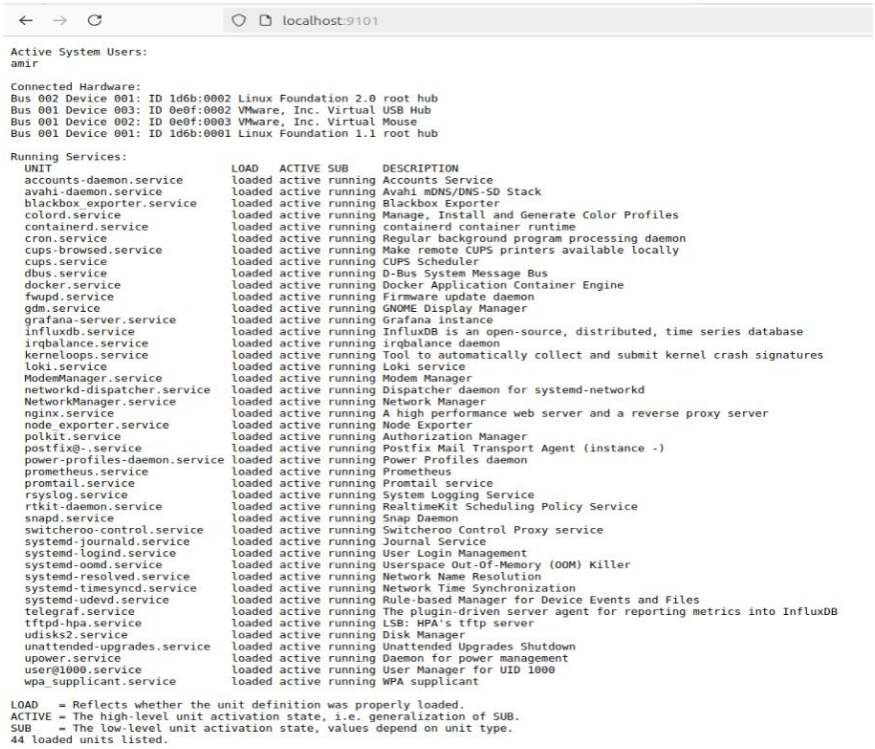
برای نصب netcat-openbsd:

```
sudo apt update
sudo apt install netcat-openbsd
```

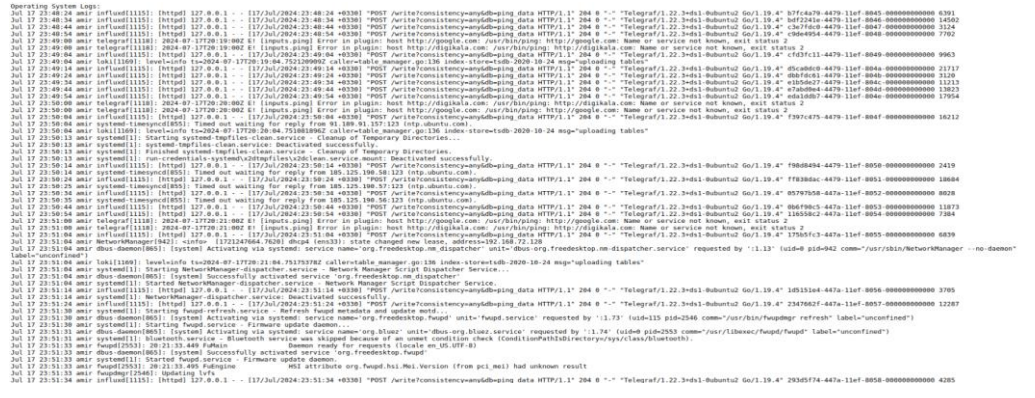
با نصب نتکت داخل فایل خود شده و پورت و دستورات اتصال را به آن اختصاص می دهیم سپس به فایل دسترسی اجرایی می دهیم و آنرا اجرا می کنیم:

```
chmod +x run_active_users_exporter.sh
./run_active_users_exporter.sh
```

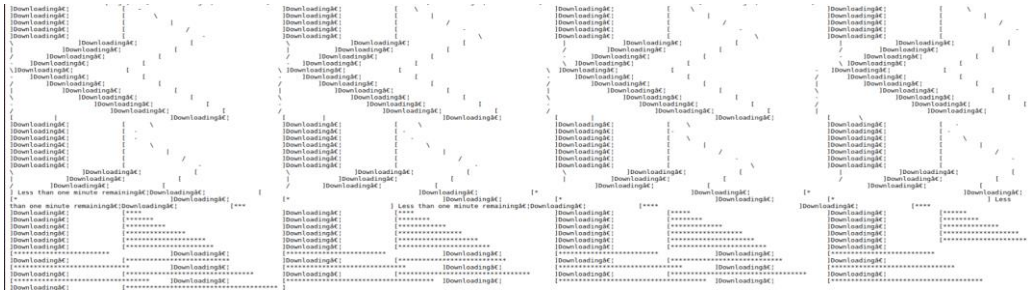
سپس می توانیم با استفاده از لوکال هاست ۹۰۹۱ که برابر با پورتی است که با فایلمان اختصاص دادیم خروجی دستورات را بر روی مرورگر هم مشاهده نماییم:



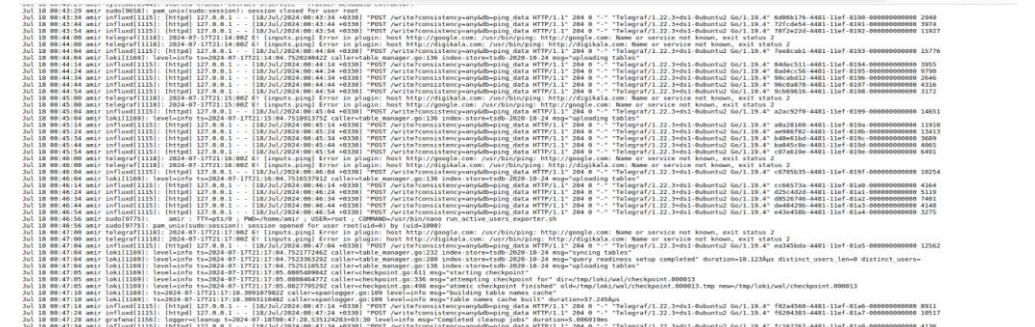
شکل ۸۷-۴: مشاهده خروجی دستورات بر روی Local Host.



شکل ۸۸-۴: مشاهده خروجی دستورات بر روی Local Host..



شکل ۸۹-۴: مشاهده خروجی دستورات بر روی Local Host..



شکل ۹۰-۴: مشاهده خروجی دستورات بر روی Local Host..

سپس زمانی که از صحت عملکرد و خروجی دستورات فایلی که ساختیم مطمئن شدیم دستورات را با دستورات قابل خواندن برای پرومتئوس جایگزین می کنیم:

```
amir@amir: ~
  GNU nano 7.2                                                    run_active_users_exporter.sh
#!/bin/bash

PORT=9101

while true; do
    {
        echo -e "HTTP/1.1 200 OK\n"

        # (میزان استفاده از هارد لحظه ای (تاریخچه ۱۵,۱۰,۵ دقیقه
        echo "# HELP disk_usage_5m میزان استفاده از هارد لحظه ای در 5 دقیقه"
        echo "# TYPE disk_usage_5m gauge"
        echo -n "disk_usage_5m "
        iostat -x 5 1 | awk 'NR==7 {print $12}'
        echo ""

        echo "# HELP disk_usage_10m میزان استفاده از هارد لحظه ای در 10 دقیقه"
        echo "# TYPE disk_usage_10m gauge"
        echo -n "disk_usage_10m "
        iostat -x 10 1 | awk 'NR==7 {print $12}'
        echo ""

        echo "# HELP disk_usage_15m میزان استفاده از هارد لحظه ای در 15 دقیقه"
        echo "# TYPE disk_usage_15m gauge"
        echo -n "disk_usage_15m "
        iostat -x 15 1 | awk 'NR==7 {print $12}'
        echo ""

        # لیست کاربران فعال
        echo "# HELP active_system_users تعداد کاربران فعال"
        echo "# TYPE active_system_users gauge"
        echo -n "active_system_users "
        who | awk '{print $1}' | sort | uniq | wc -l
        echo ""

        # لیست سخت افزارهای متصل
        echo "# HELP connected_hardware تعداد سخت افزارهای متصل"
        echo "# TYPE connected_hardware gauge"
        echo -n "connected_hardware "
        lsusb | wc -l
        echo ""

        # لیست سرویس‌های در حال اجرا
        echo "# HELP running_services تعداد سرویس‌های در حال اجرا"
        echo "# TYPE running_services gauge"
        echo -n "running_services "
        systemctl list-units --type=service --state=running | grep -v '\-.mount' | wc -l
        echo ""

        # لاگ‌های سطح سیستم عامل
        echo "# HELP operating_system_logs تعداد لاگ‌های سیستم عامل"
        echo "# TYPE operating_system_logs gauge"
        echo -n "operating_system_logs "
        journalctl --since "1 hour ago" | wc -l
        echo ""
    } | nc -l -p $PORT -q 1
done
```

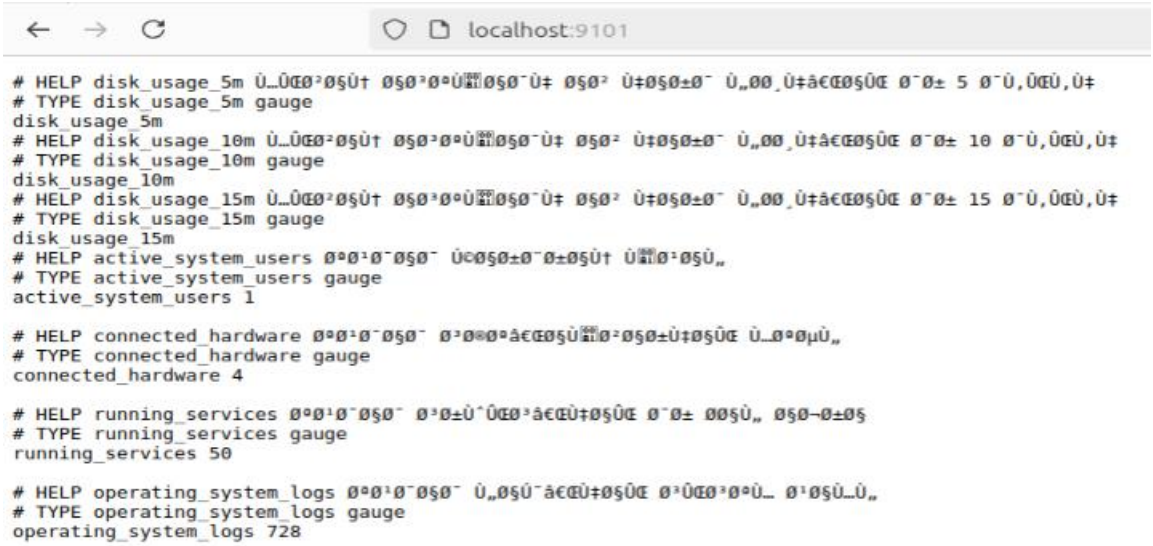
شکل ۹۱-۴: جایگزینی دستورات با دستورات قابل فهم برای پرومتئوس.

سپس فایل را مجددا اجرا می کنیم:

```
amir@amir:~$ ./run_active_users_exporter.sh
```

شکل ۹۲-۴: اجرای مجدد فایل.

حالا، اگر مرورگر خود را باز کنید و به http://localhost:۹۱۰۱ بروید، باید اطلاعات مورد نظر را مشاهده کنید.

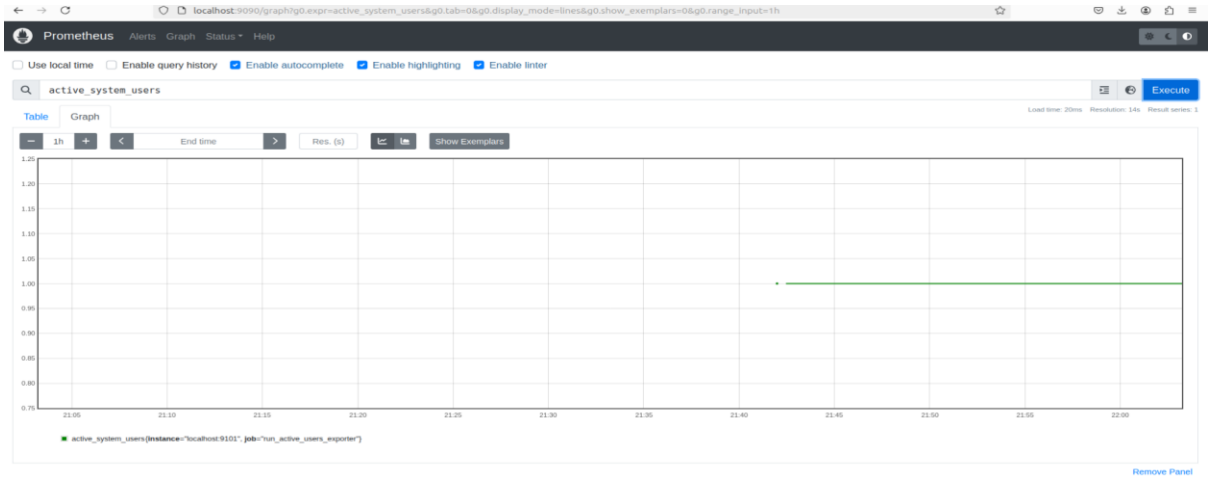


شکل ۹۳-۴: تست دستورات قابل فهم برای پرومتئوس.

حال با استفاده از دستوراتی که تعریف کردیم خروجی لازم برای موارد خواسته شده در گرافانا و پرومتئوس در دسترس هستند.

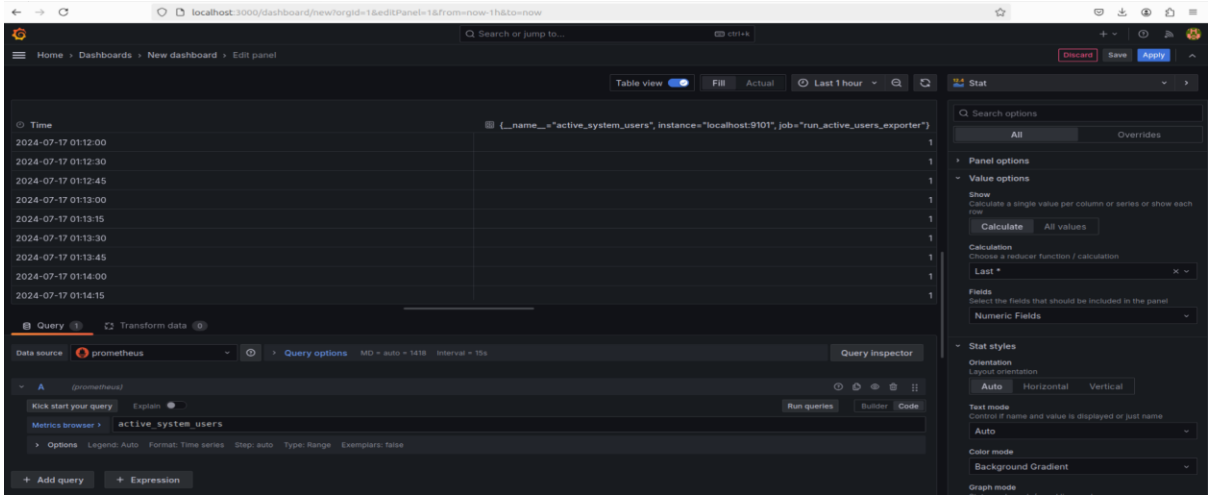
۷-۲-۴- لیست کاربران فعال سیستم:

خروجی در Prometheus:

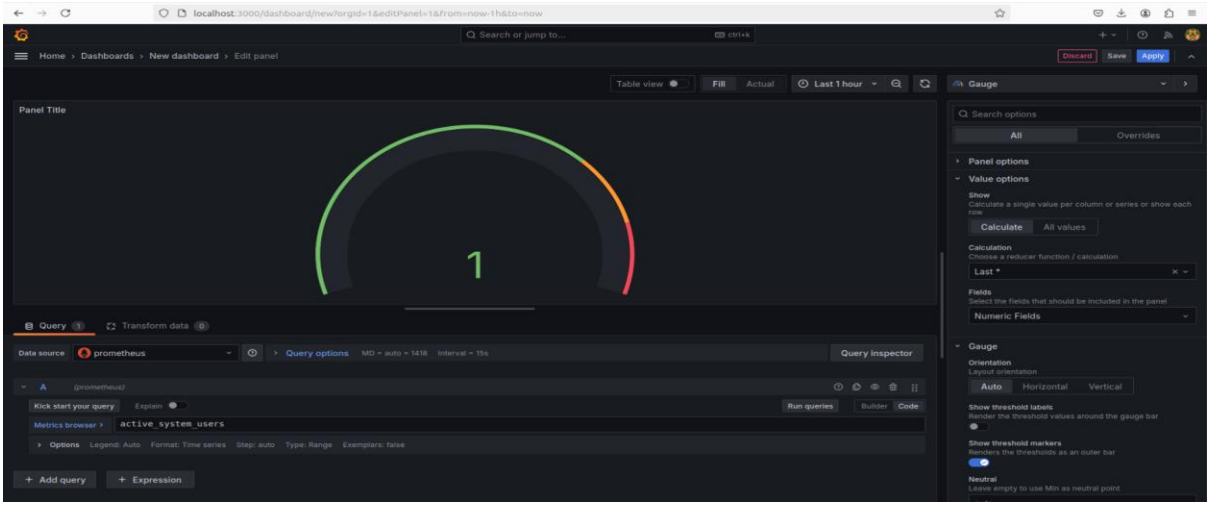


شکل ۹۴-۴: لیست کاربران فعال سیستم (Prometheus).

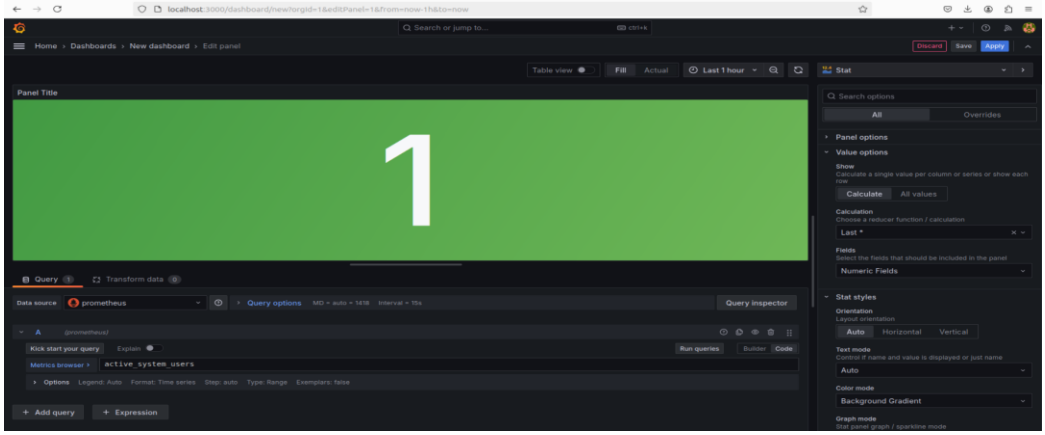
خروجی در Grafana:



شکل ۹۵-۴: لیست کاربران فعال سیستم (Grafana).

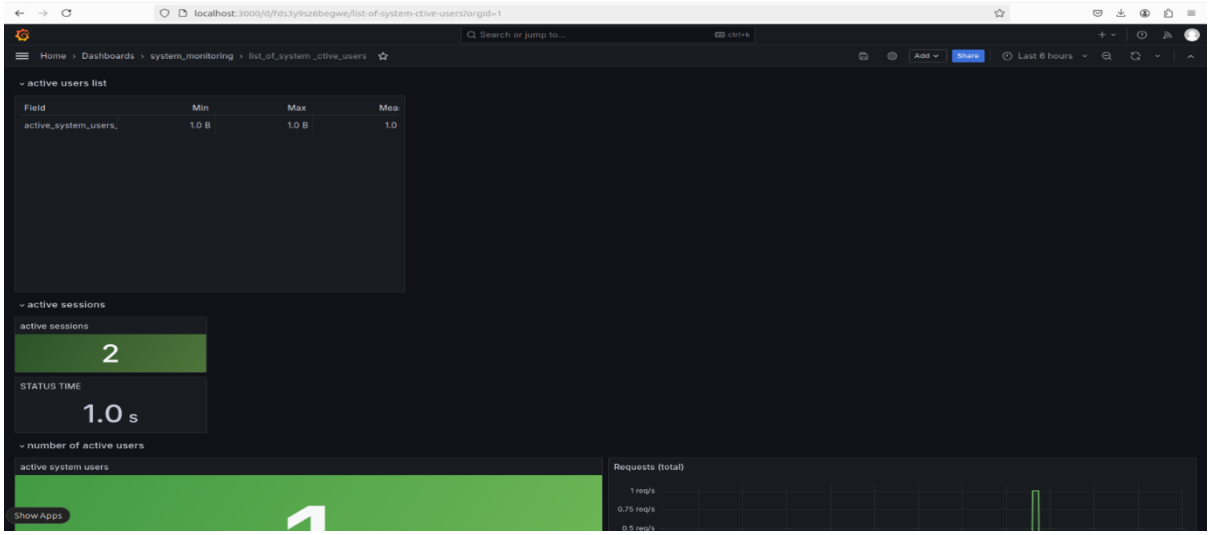


شکل ۹۶-۴: لیست کاربران فعال سیستم (Grafana).

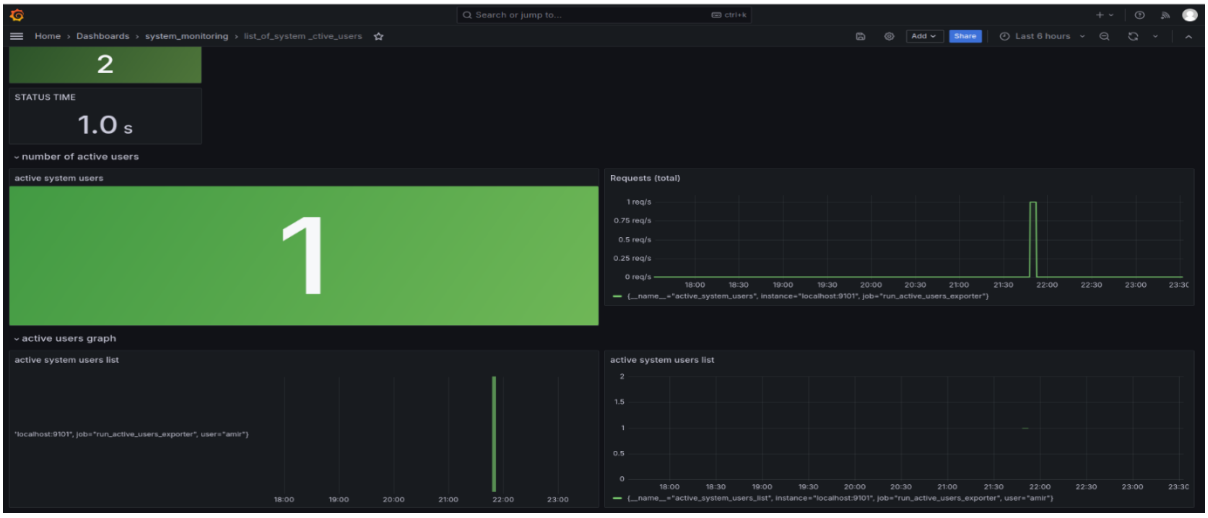


شکل ۹۷-۴: لیست کاربران فعال سیستم (Grafana).

خروجی نهایی داشبورد:



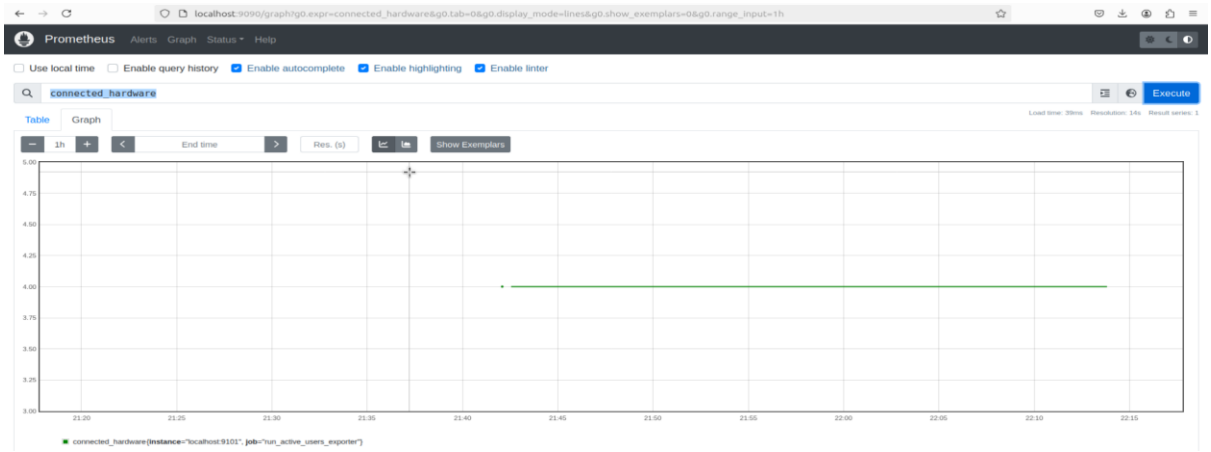
شکل ۹۸-۴: تصویر کامل داشبورد گرافیکی لیست کاربران فعال سیستم (Grafana).



شکل ۹۹-۴: تصویر کامل داشبورد گرافیکی لیست کاربران فعال سیستم (Grafana).

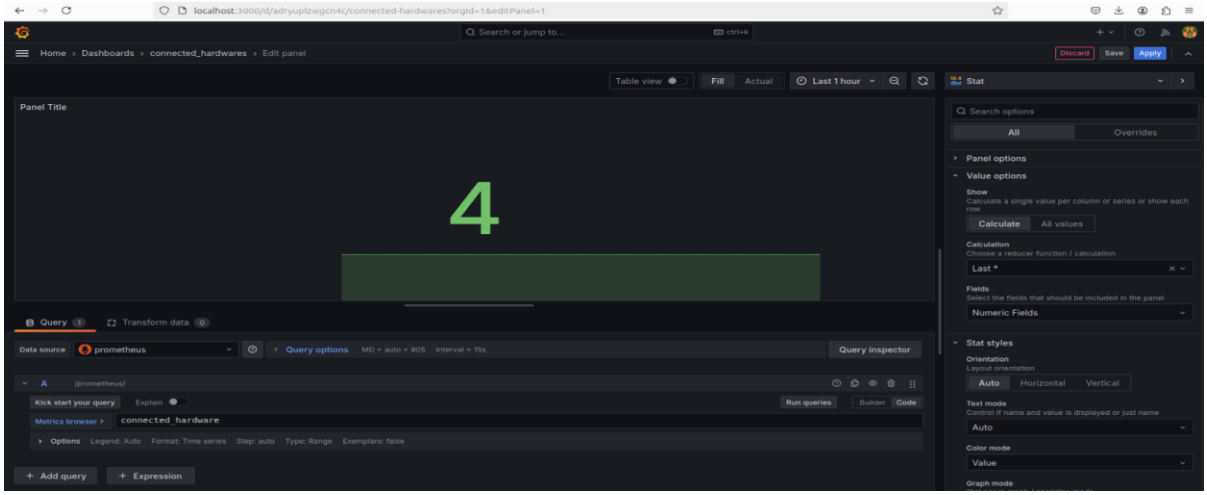
۸-۲-۴- لیست سخت افزار های مختلف:

خروجی در Prometheus :



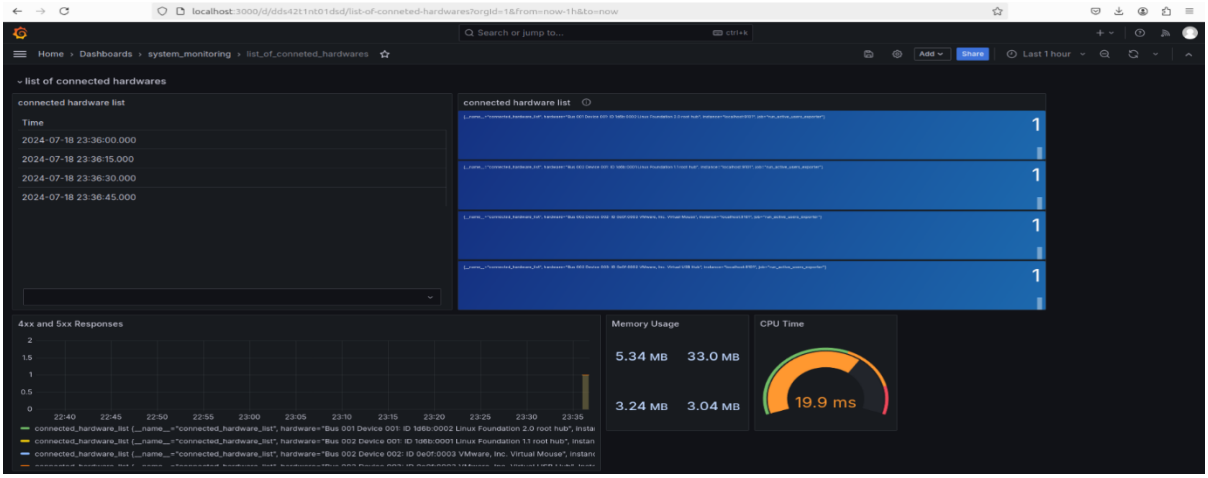
شکل ۱۰۰-۴: لیست سخت افزار های مختلف (Prometheus).

خروجی در Grafana :

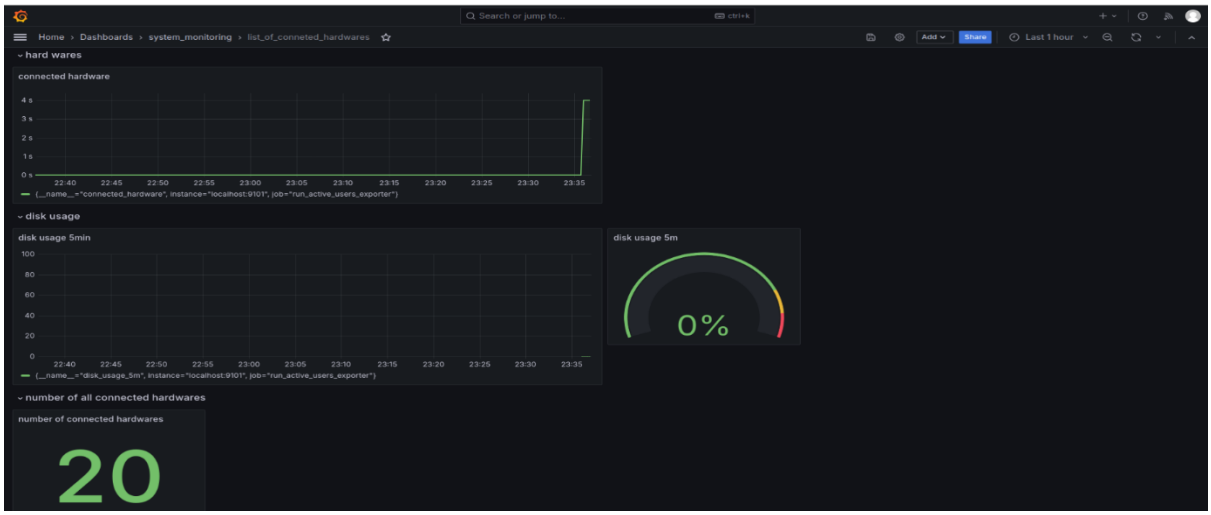


شکل ۱۰۱-۴: لیست سخت افزار های مختلف (Grafana).

خروجی نهایی داشبورد:



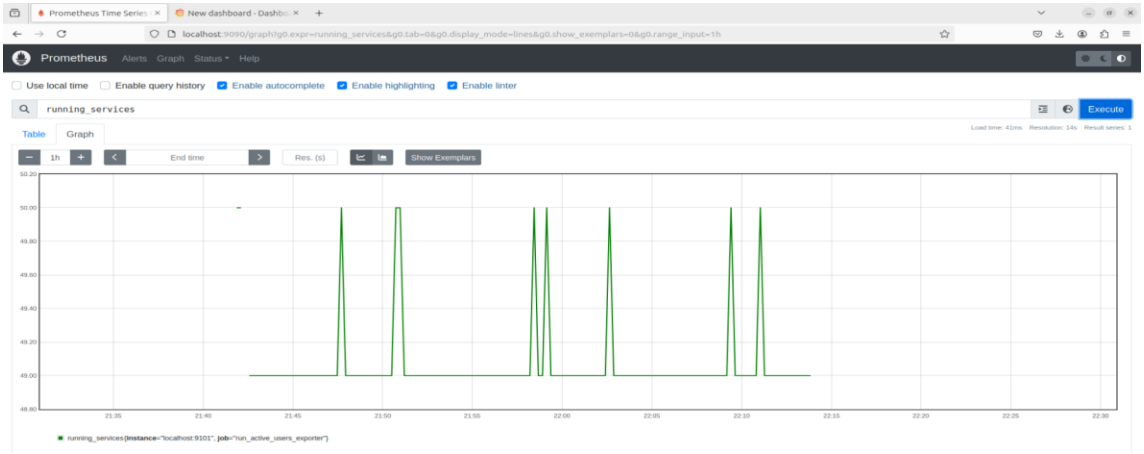
شکل ۱۰۲-۴: تصویر کامل داشبورد گرافیکی لیست سخت افزار های مختلف (Grafana).



شکل ۱۰۳-۴: تصویر کامل داشبورد گرافیکی لیست سخت افزار های مختلف (Grafana).

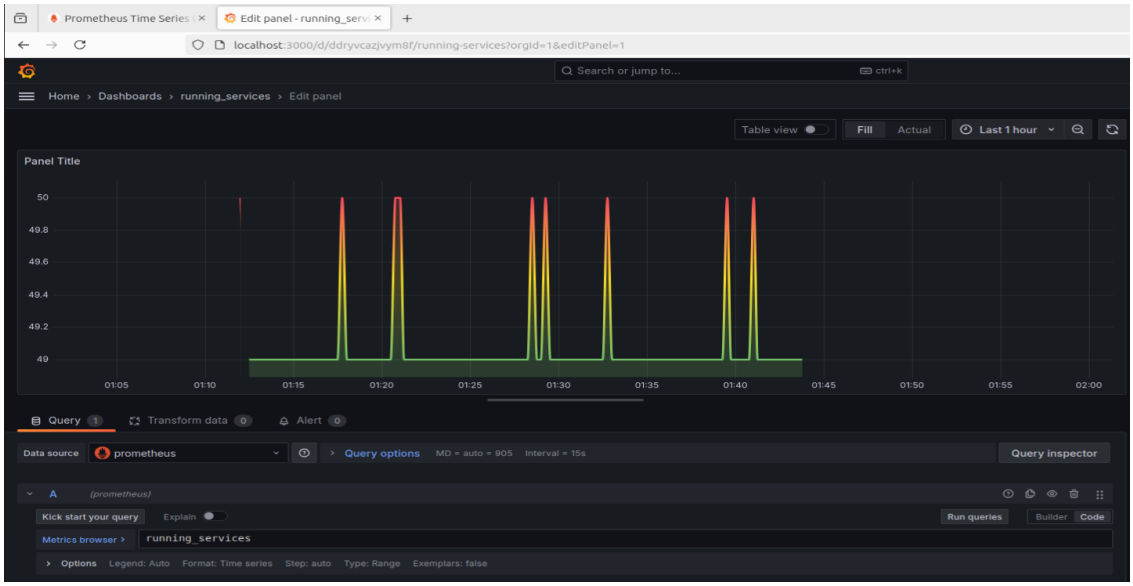
۹-۲-۴- سرویس های درحال اجرا :

خروجی در Prometheus :



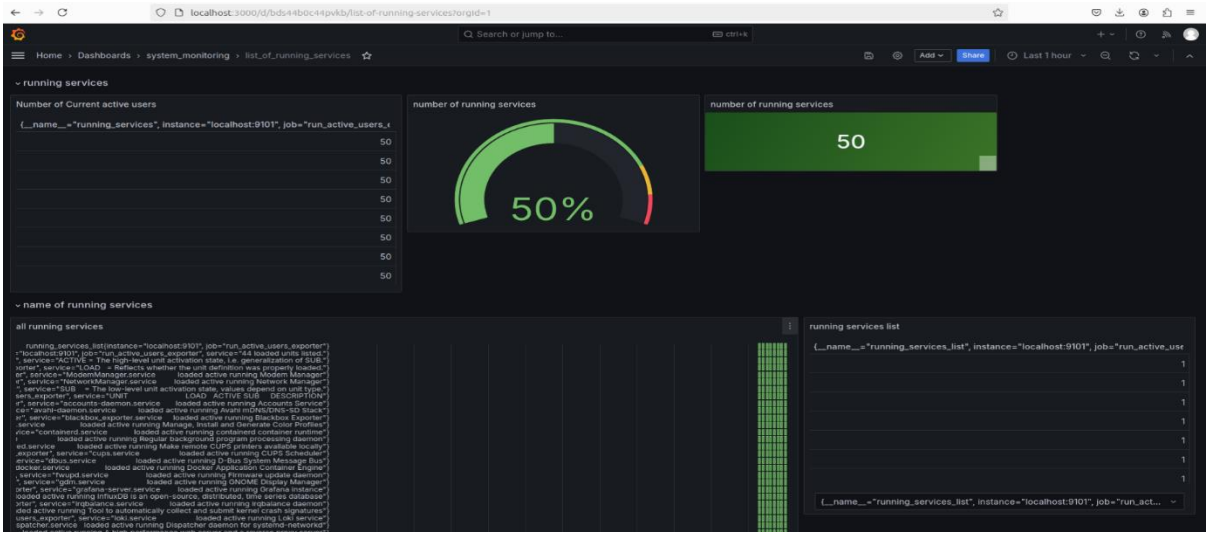
شکل ۱۰۴-۴: سرویس های درحال اجرا (Prometheus).

خروجی در Grafana :

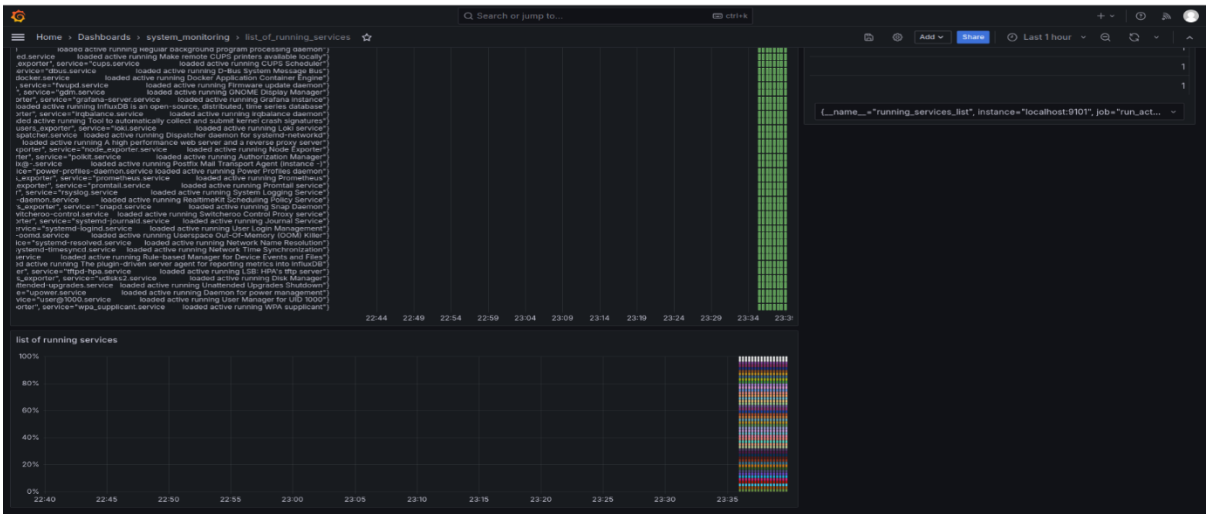


شکل ۱۰۵-۴: سرویس های درحال اجرا (Grafana).

خروجی نهایی داشبورد:



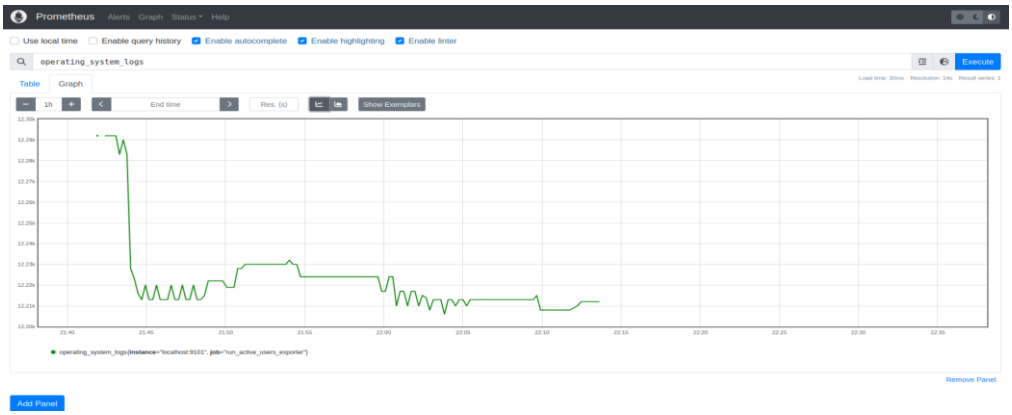
شکل ۱۰۶-۴: تصویر کامل داشبورد گرافیکی سرویس های درحال اجرا (Grafana).



شکل ۱۰۷-۴: تصویر کامل داشبورد گرافیکی سرویس های درحال اجرا (Grafana).

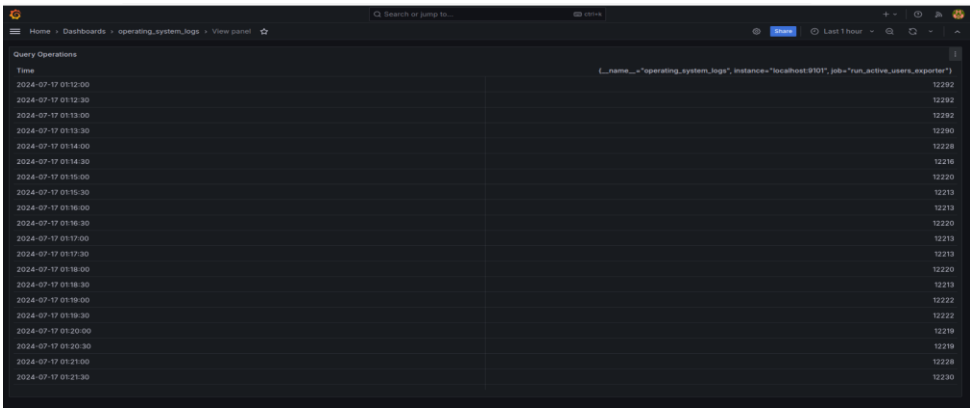
۱۰-۲-٤- لاگ های سطح سیستم عامل :

خروجی در Prometheus :



شکل ۱۰۸-۴: لاگ های سطح سیستم عامل (Prometheus).

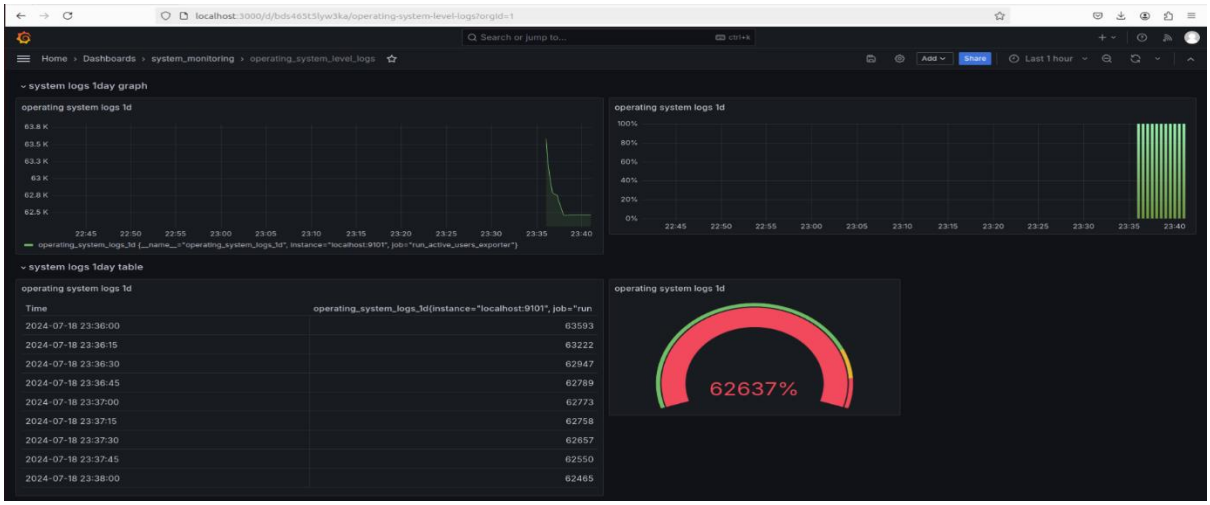
خروجی در Grafana :



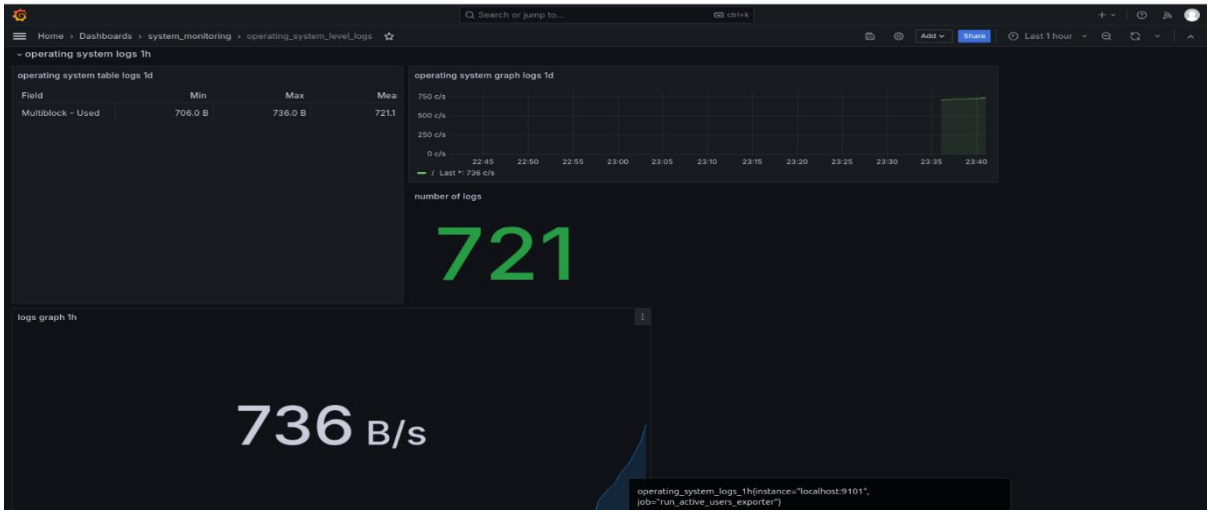
شکل ۱۰۹-۴: لاگ های سطح سیستم عامل (Grafana).

شکل ۱۱۰-۴: لاگ های سطح سیستم عامل (Grafana).

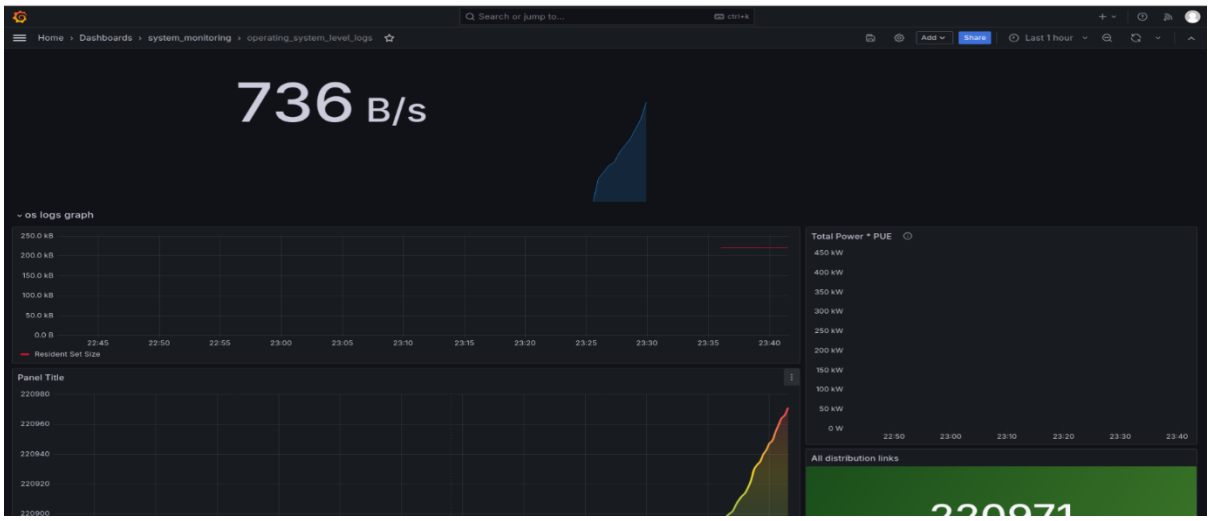
خروجی نهایی داشبورد:



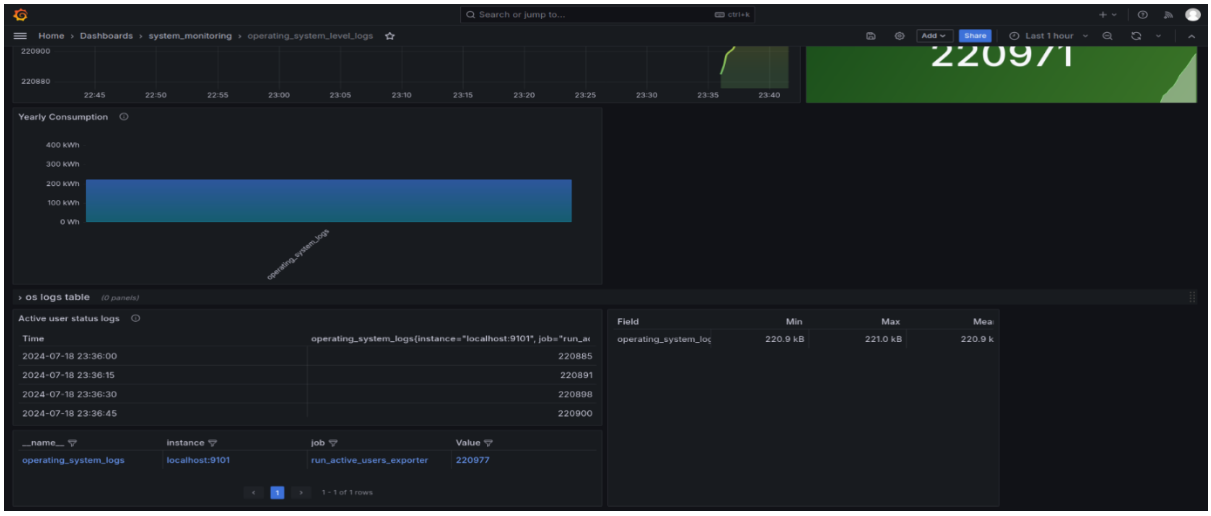
شکل ۱۱۱-۴: تصویر کامل داشبورد گرافیکی لاگ های سطح سیستم عامل (Grafana).



شکل ۱۱۲-۴: تصویر کامل داشبورد گرافیکی لاگ های سطح سیستم عامل (Grafana).



شکل ۱۱۳-۴: تصویر کامل داشبورد گرافیکی لاگ های سطح سیستم عامل (Grafana).



شکل ۱۱۴-۴: تصویر کامل داشبورد گرافیکی لاگ های سطح سیستم عامل (Grafana).

۳-۴- مانیتورینگ وب سایت

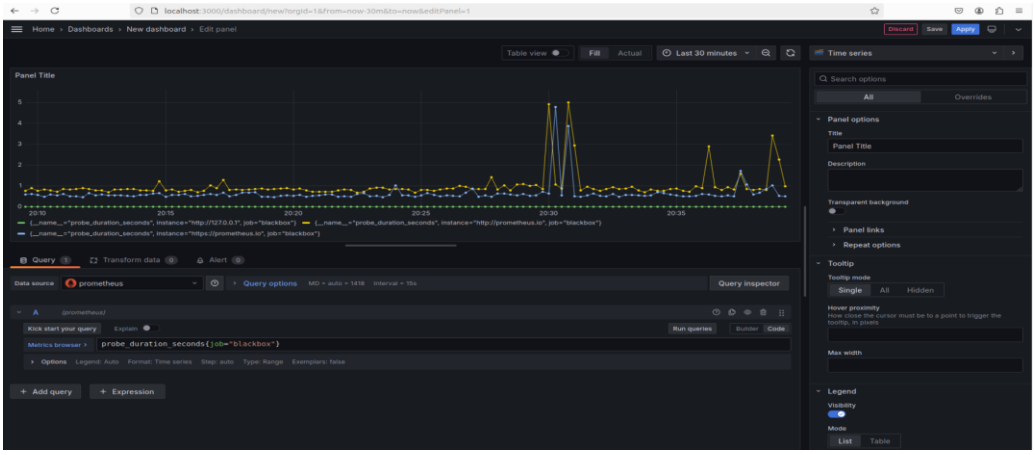
۱-۳-۴- مدت زمان لود یک صفحه:

لحظه ای:

خروجی در Prometheus:

شکل ۱۱۵-۴: مدت زمان لود یک صفحه لحظه ای (Prometheus).

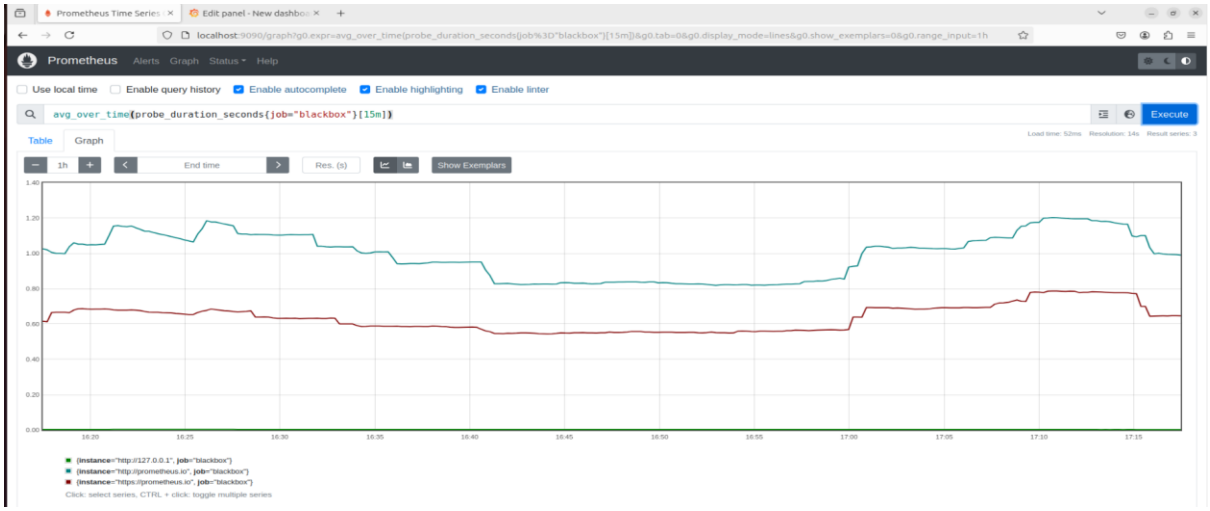
خروجی در Grafana:



شکل ۱۱۶-۴: مدت زمان لود یک صفحه لحظه ای (Grafana).

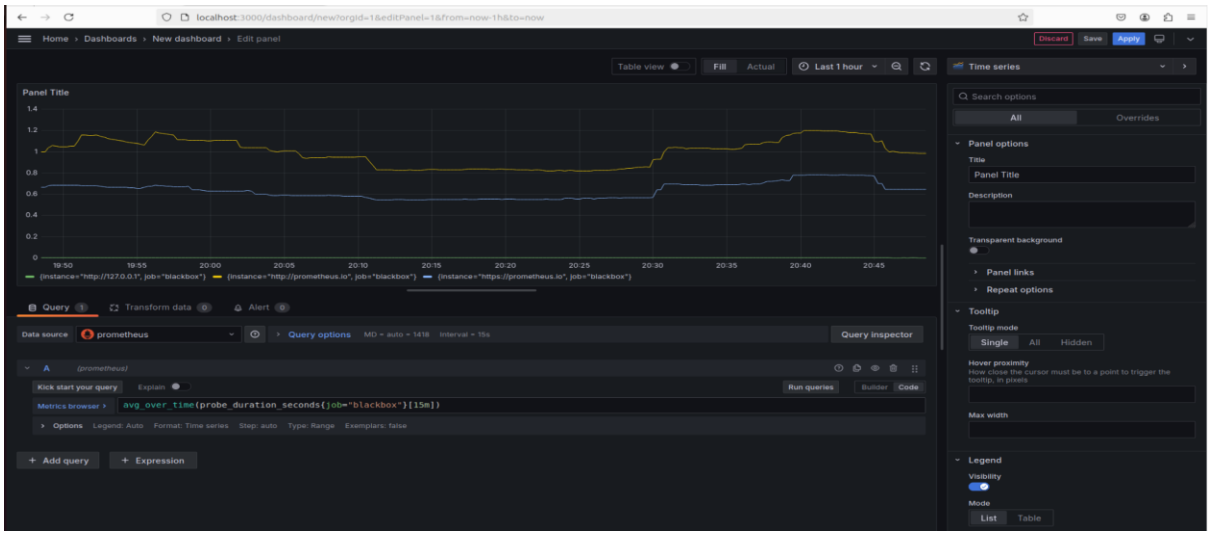
۱۵ دقیقه:

خروجی در Prometheus :



شکل ۱۱۷-۴: مدت زمان لود یک صفحه ۱۵ دقیقه (Prometheus).

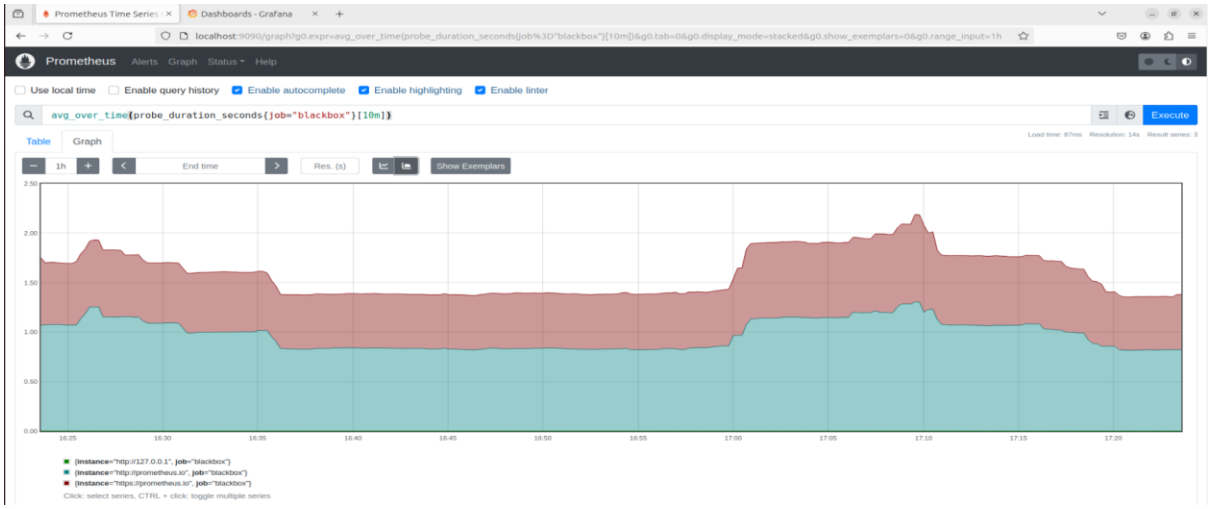
خروجی در Grafana :



شکل ۱۱۸-۴: مدت زمان لود یک صفحه ۱۵ دقیقه (Grafana).

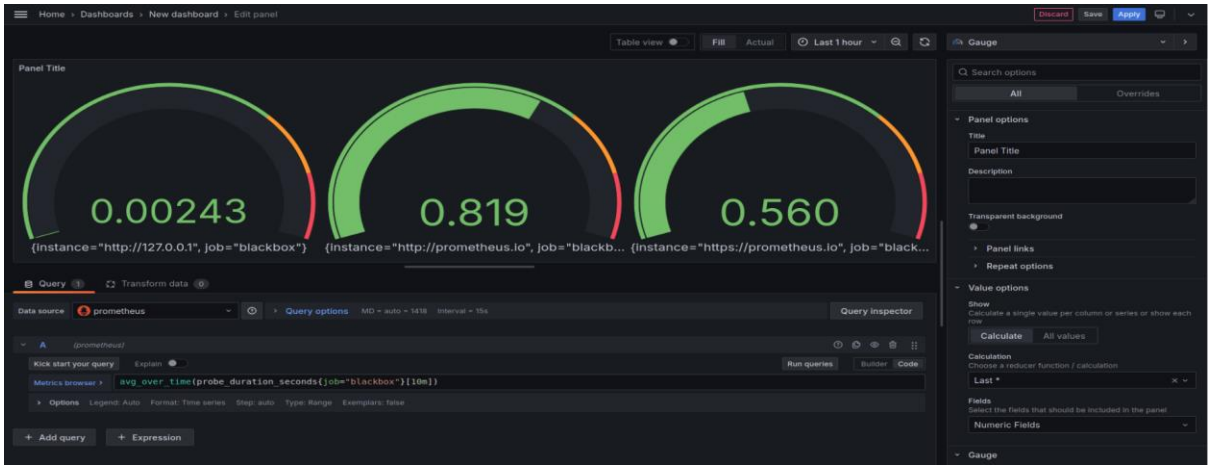
۱۰ دقیقه:

خروجی در Prometheus:



شکل ۱۱۹-۴: مدت زمان لود یک صفحه ۱۰ دقیقه (Prometheus).

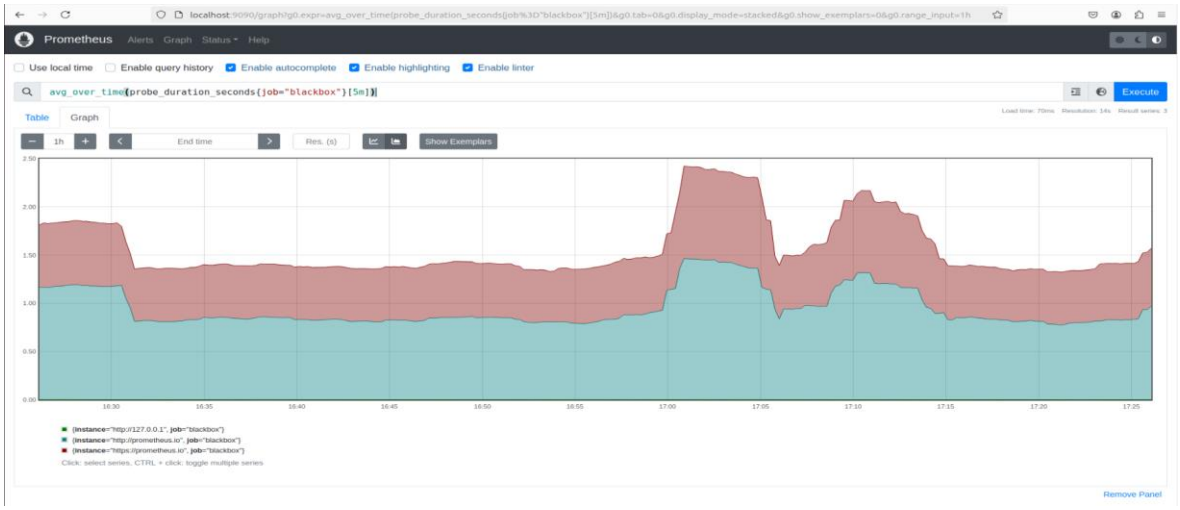
خروجی در Grafana:



شکل ۱۲۰-۴: مدت زمان لود یک صفحه ۱۰ دقیقه (Grafana).

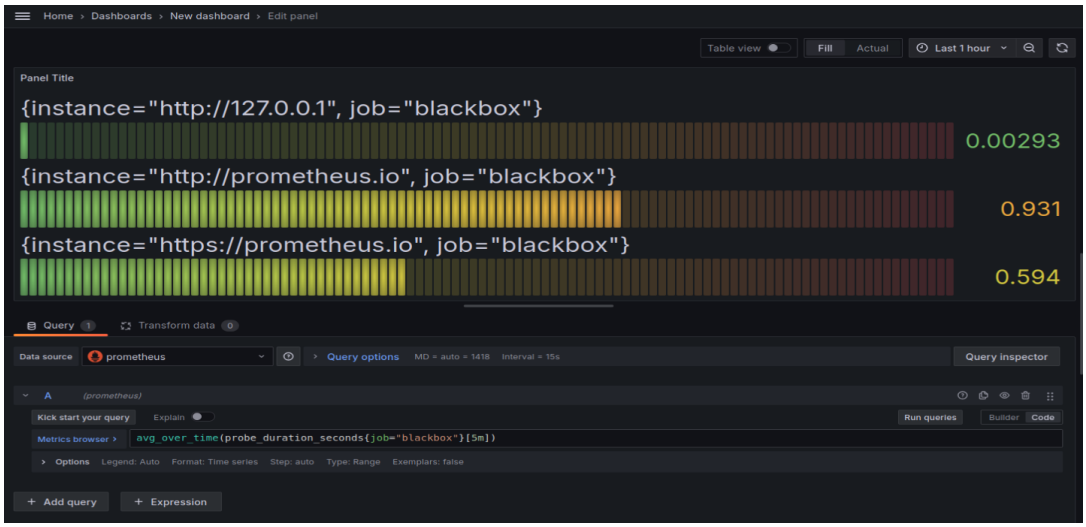
۵ دقیقه:

خروجی در Prometheus:



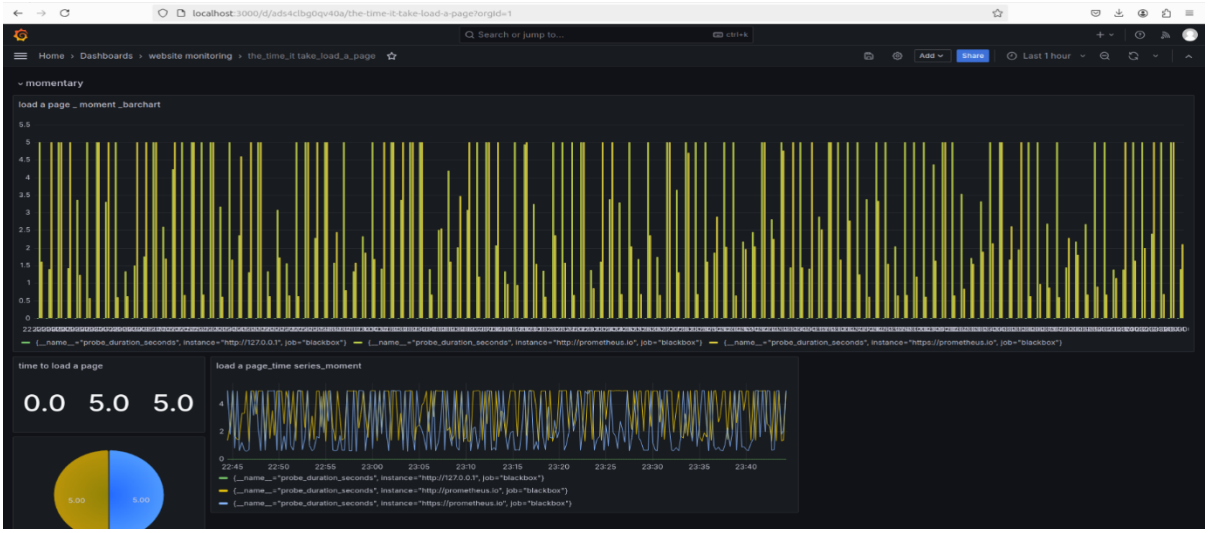
شکل ۱۲۱-۴: مدت زمان لود یک صفحه ۵ دقیقه (Prometheus).

خروجی در Grafana:

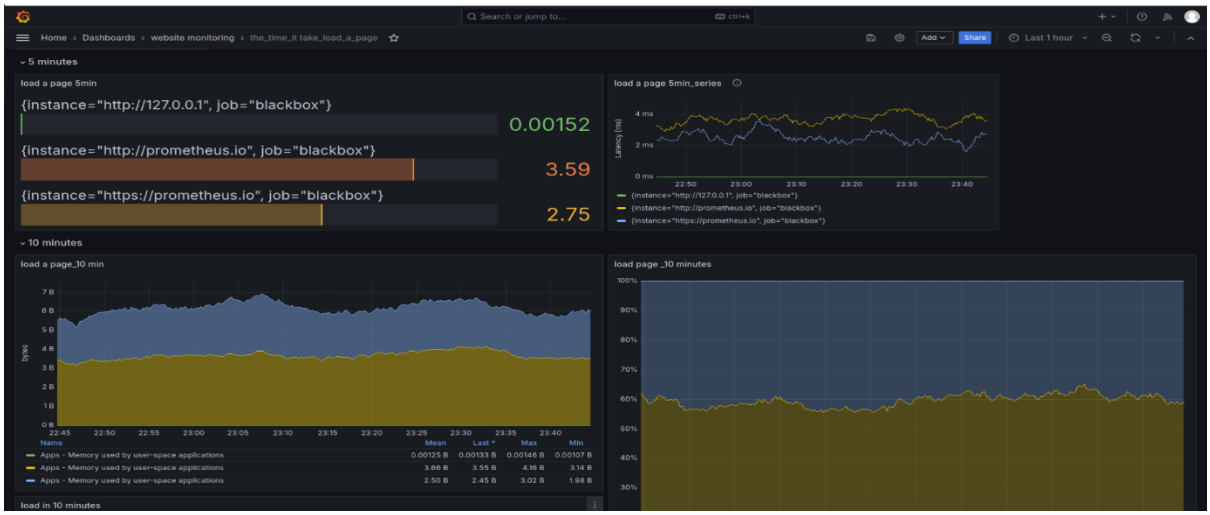


شکل ۱۲۲-۴: مدت زمان لود یک صفحه ۵ دقیقه (Grafana).

خروجی نهایی داشبورد:



شکل ۱۲۳-۴: تصویر کامل داشبورد گرافیکی مدت زمان لود یک صفحه (Grafana).



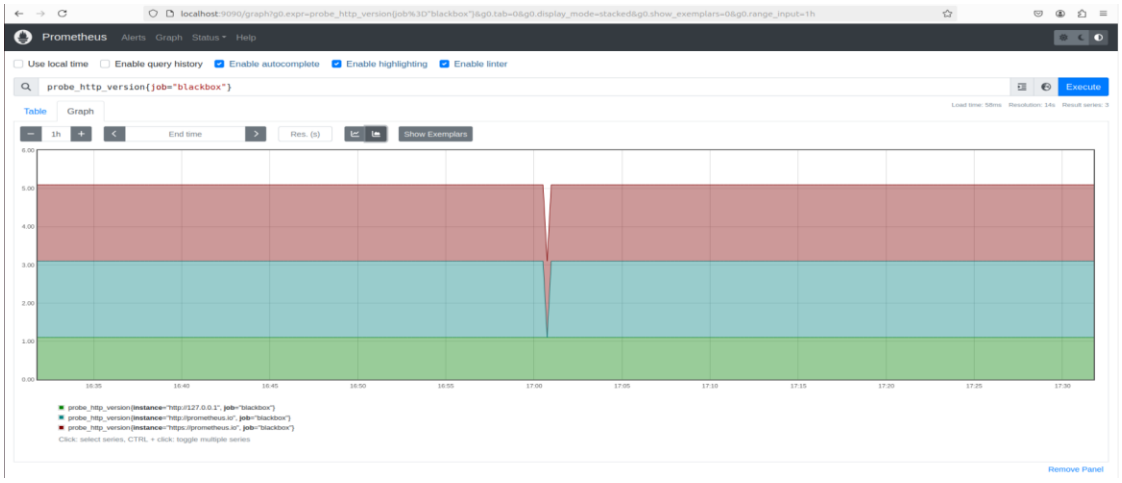
شکل ۱۲۴-۴: تصویر کامل داشبورد گرافیکی مدت زمان لود یک صفحه (Grafana).

شکل ۱۲۵-۴: تصویر کامل داشبورد گرافیکی مدت زمان لود یک صفحه (Grafana).

شکل ۱۲۶-۴: تصویر کامل داشبورد گرافیکی مدت زمان لود یک صفحه (Grafana).

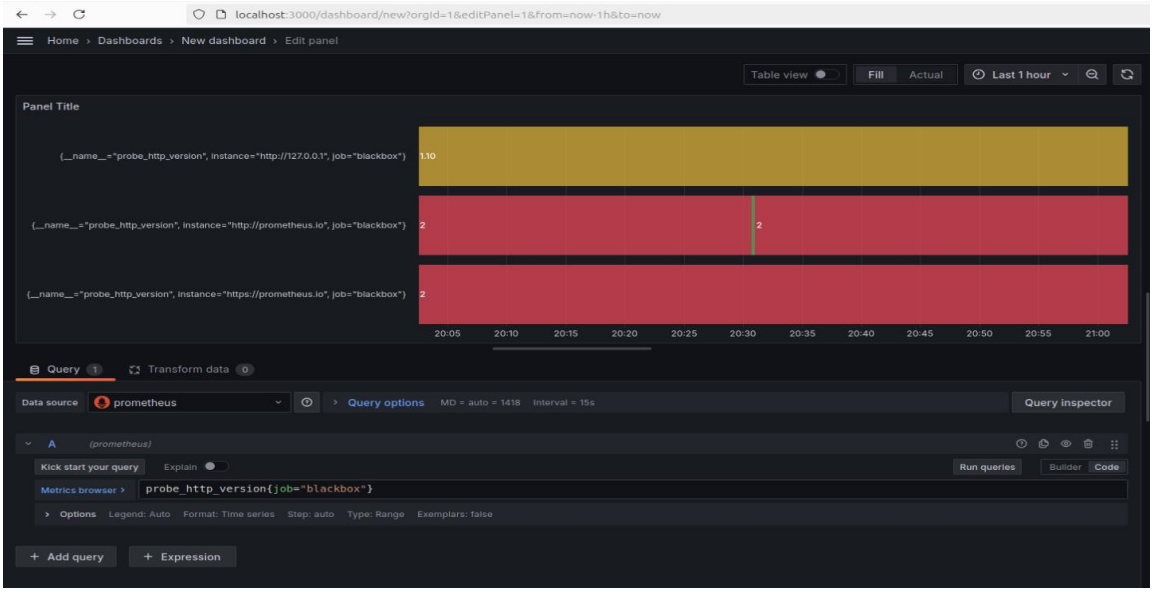
۲-۳-۴- نوع و نام وب سرور:

خروجی در Prometheus:



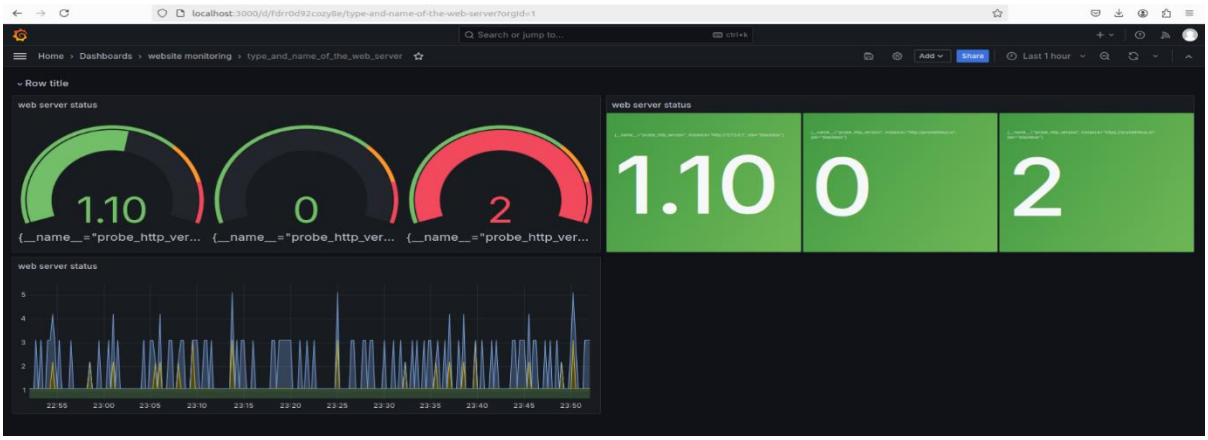
شکل ۱۲۷-۴: نوع و نام وب سرور (Prometheus).

خروجی در Grafana:



شکل ۱۲۸-۴: نوع و نام وب سرور (Grafana).

خروجی نهایی داشبورد:

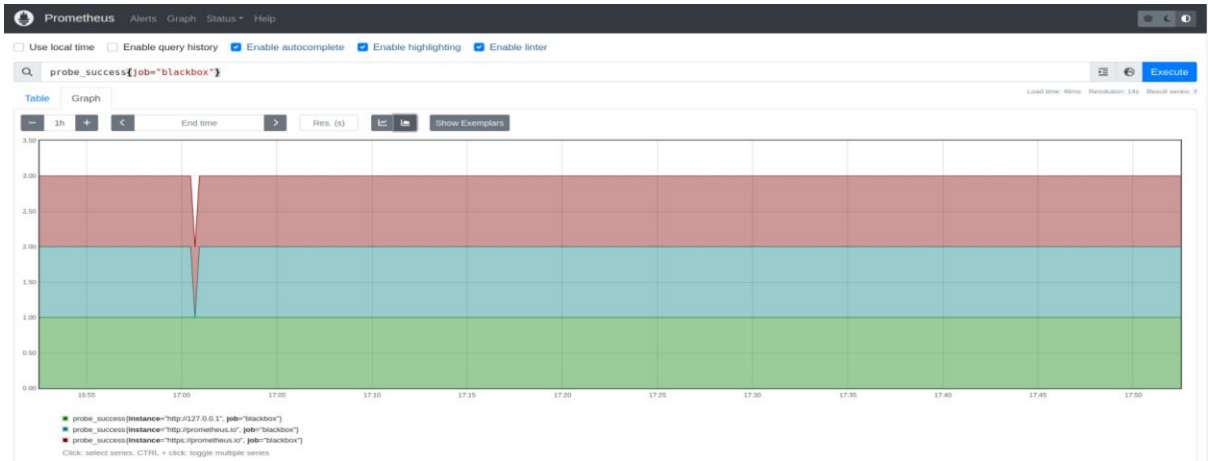


شکل ۱۲۹-۴: تصویر کامل داشبورد گرافیکی نوع و نام وب سرور (Grafana).

۳-۳-۴- وضعیت آنلاین / آفلاین صفحه:

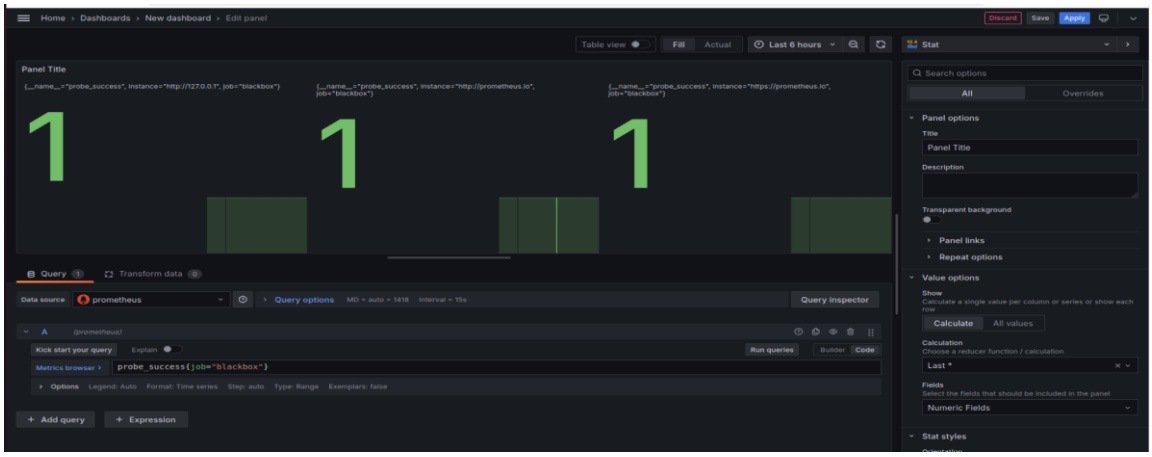
لحظه ای:

خروجی در Prometheus:



شکل ۱۳۰-۴: وضعیت آنلاین / آفلاین صفحه لحظه ای (Prometheus).

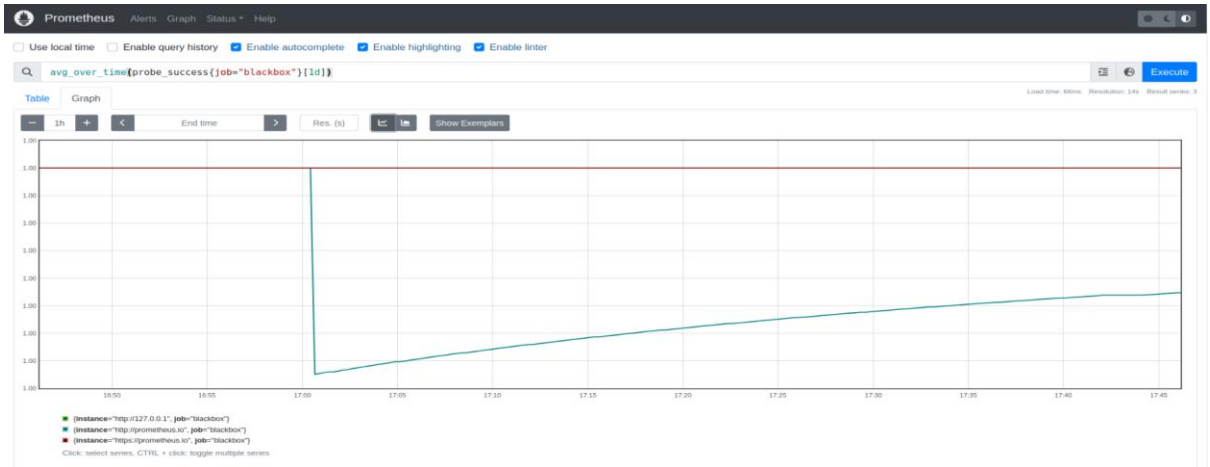
خروجی در Grafana:



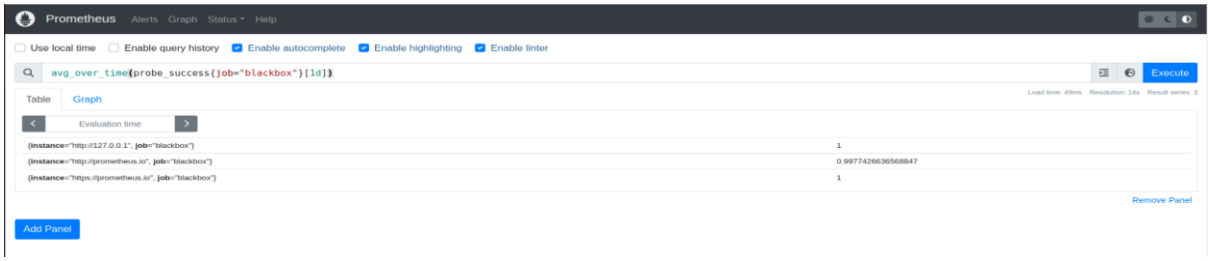
شکل ۱۳۱-۴: وضعیت آنلاین / آفلاین صفحه لحظه ای (Grafana).

روزانه:

خروجی در Prometheus :

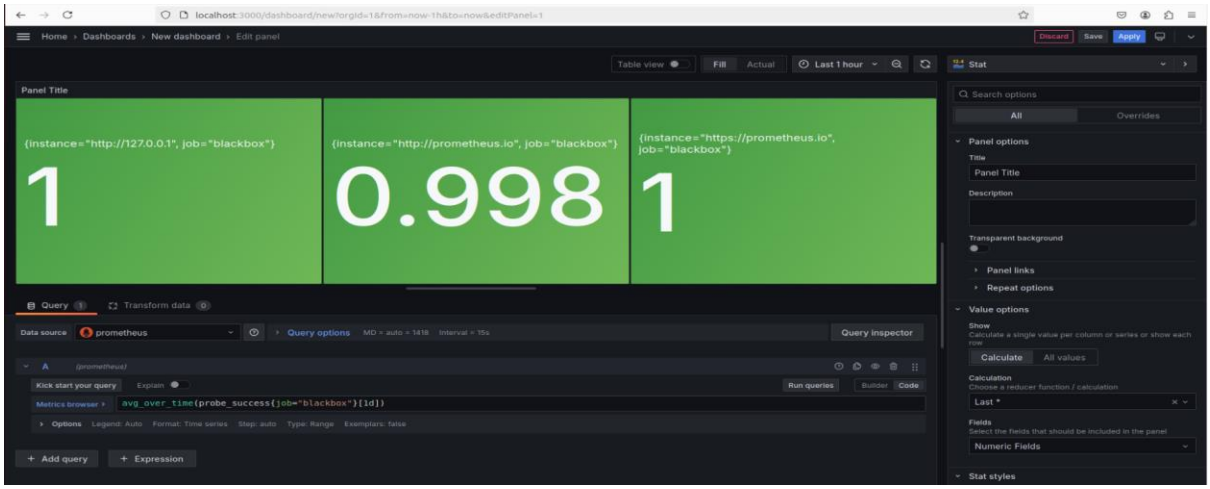


شکل ۱۳۲-۴: وضعیت آنلاین / آفلاین صفحه روزانه (Prometheus).

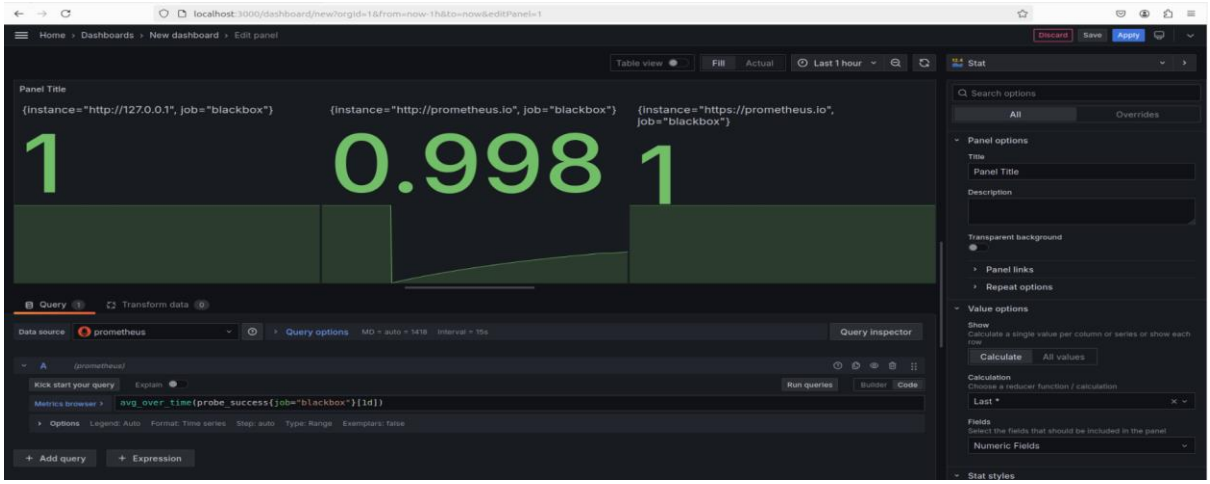


شکل ۱۳۳-۴: وضعیت آنلاین / آفلاین صفحه روزانه (Prometheus).

خروجی در Grafana:



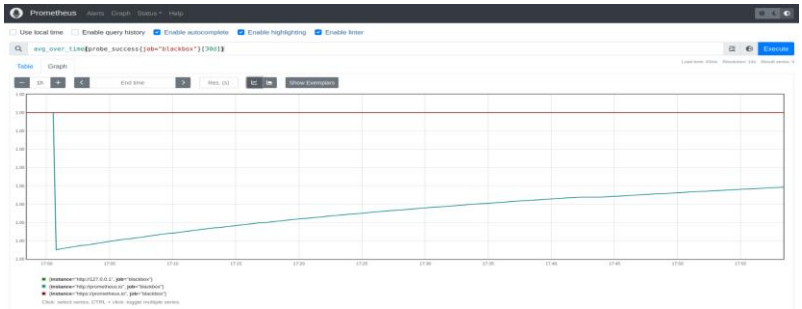
شکل ۱۳۴-۴: وضعیت آنلاین / آفلاین صفحه روزانه (Grafana).



شکل ۱۳۵-۴: وضعیت آنلاین / آفلاین صفحه روزانه (Grafana).

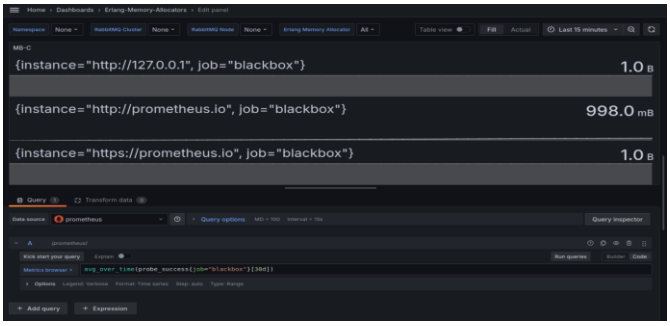
ماهانه:

خروجی در Prometheus :

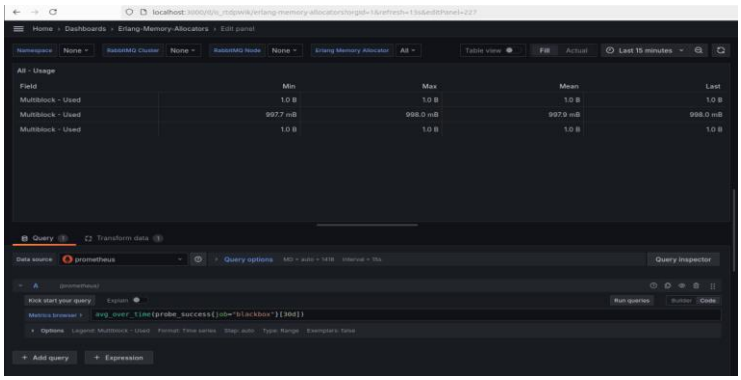


شکل ۱۳۶-۴: وضعیت آنلاین / آفلاین صفحه ماهانه (Prometheus).

خروجی در Grafana :



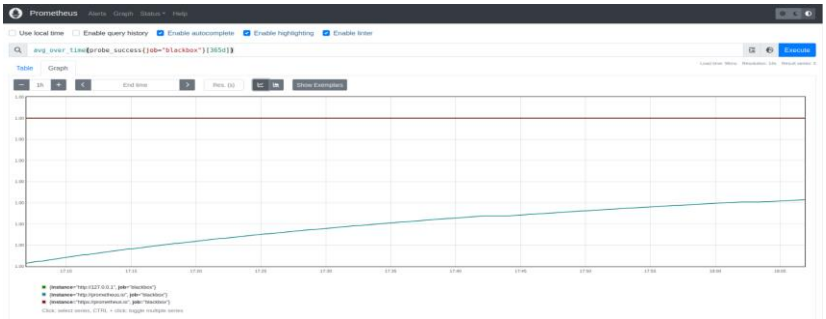
شکل ۱۳۷-۴: وضعیت آنلاین / آفلاین صفحه ماهانه (Grafana).



شکل ۱۳۸-۴: وضعیت آنلاین / آفلاین صفحه روزانه (Grafana).

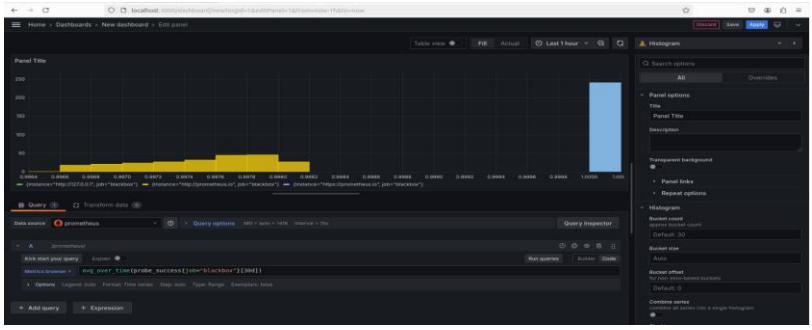
سالانه:

خروجی در Prometheus:

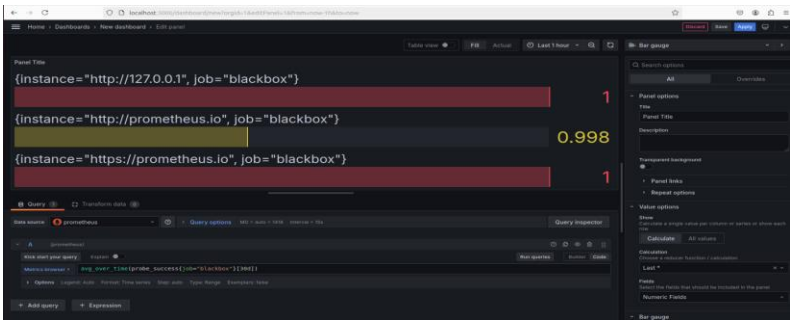


شکل ۱۳۹-۴: وضعیت آنلاین / آفلاین صفحه سالانه (Prometheus).

خروجی در Grafana:

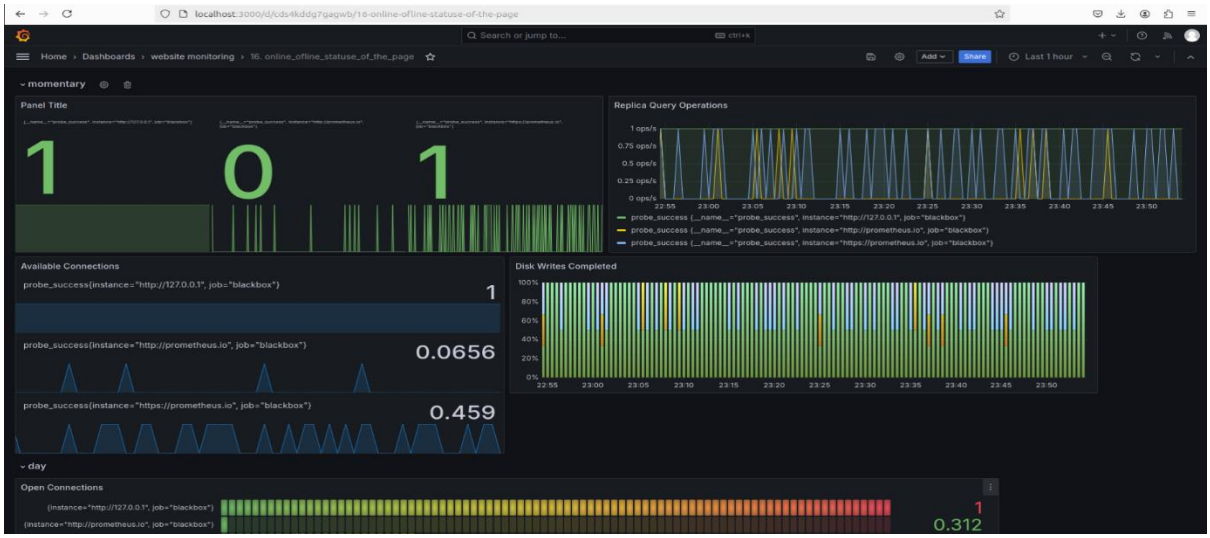


شکل ۱۴۰-۴: وضعیت آنلاین / آفلاین صفحه سالانه (Grafana).

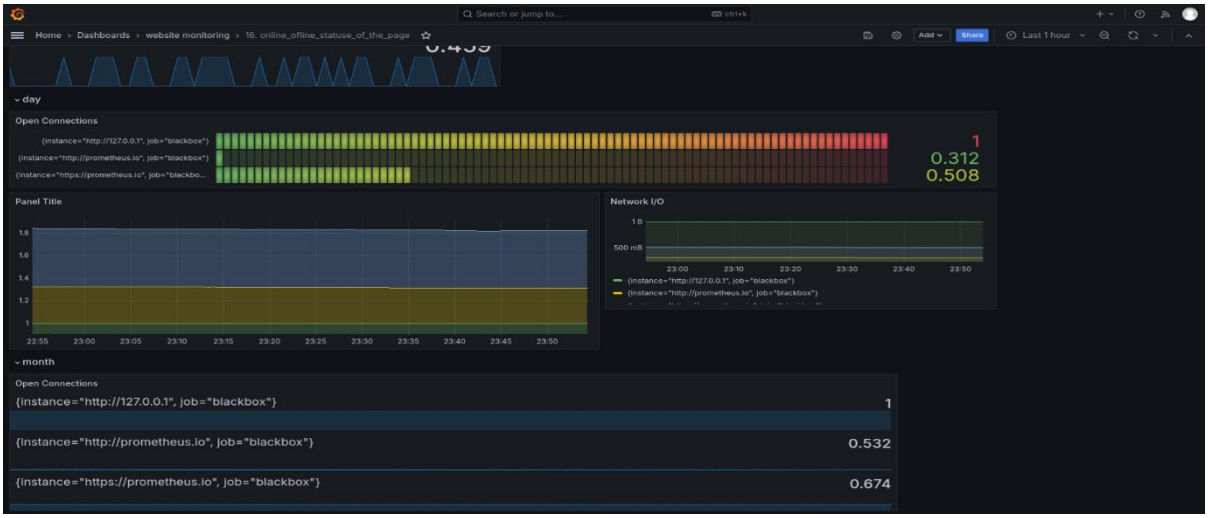


شکل ۱۴۱-۴: وضعیت آنلاین / آفلاین صفحه سالانه (Prometheus).

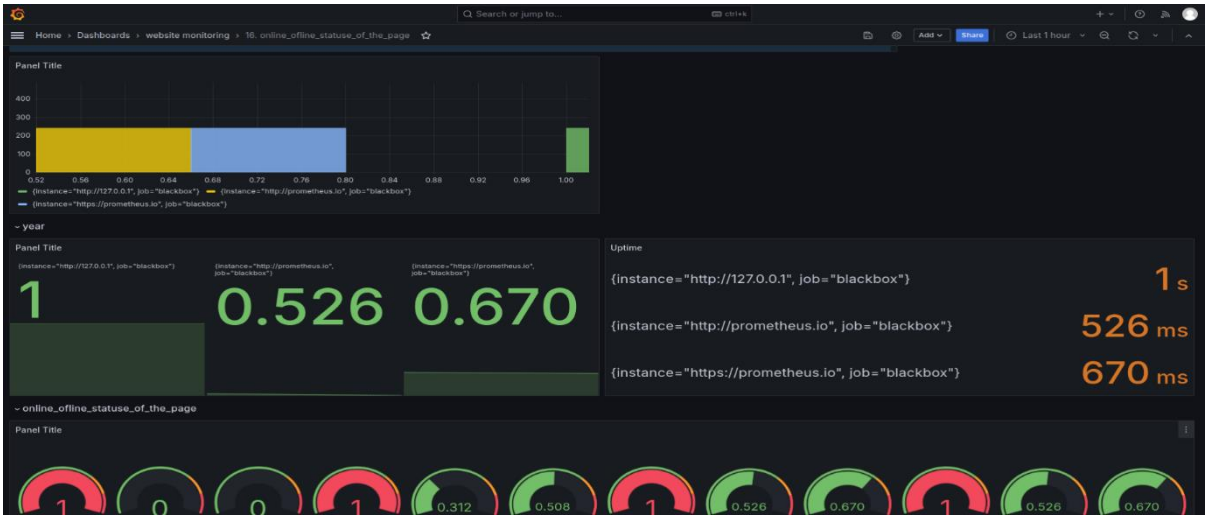
خروجی نهایی داشبورد:



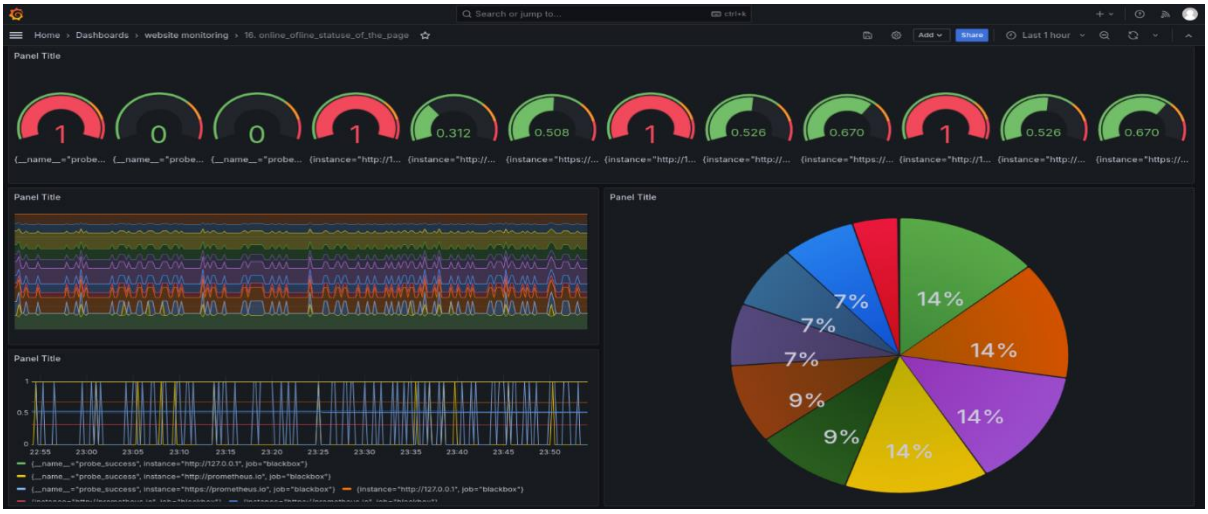
شکل ۱۴۲-۴: تصویر کامل داشبورد وضعیت آنلاین / آفلاین صفحه (Grafana).



شکل ۱۴۳-۴: تصویر کامل داشبورد وضعیت آنلاین / آفلاین صفحه (Grafana).



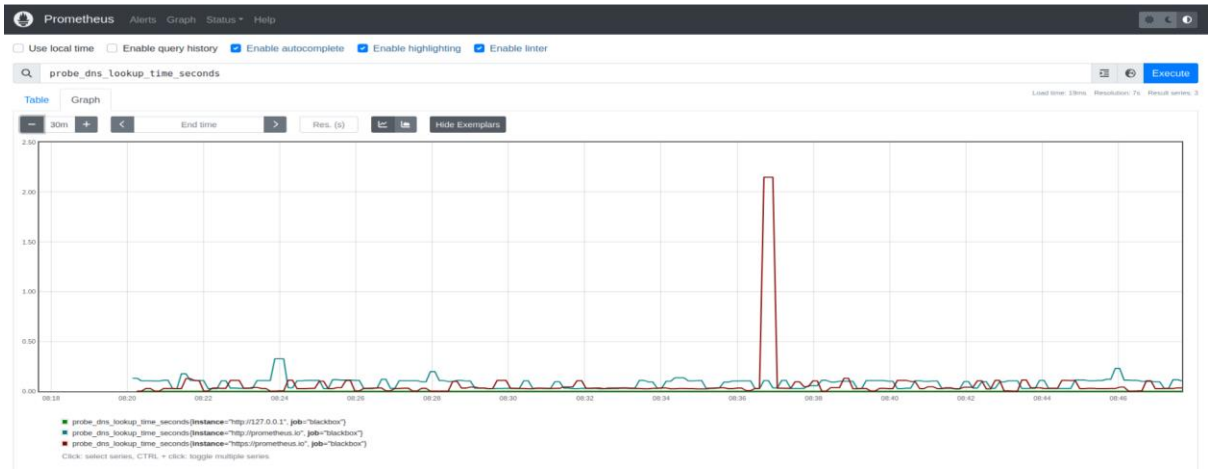
شکل ۱۴۴-۴: تصویر کامل داشبورد وضعیت آنلاین / آفلاین صفحه (Grafana).



شکل ۱۴۵-۴: تصویر کامل داشبورد وضعیت آنلاین / آفلاین صفحه (Grafana).

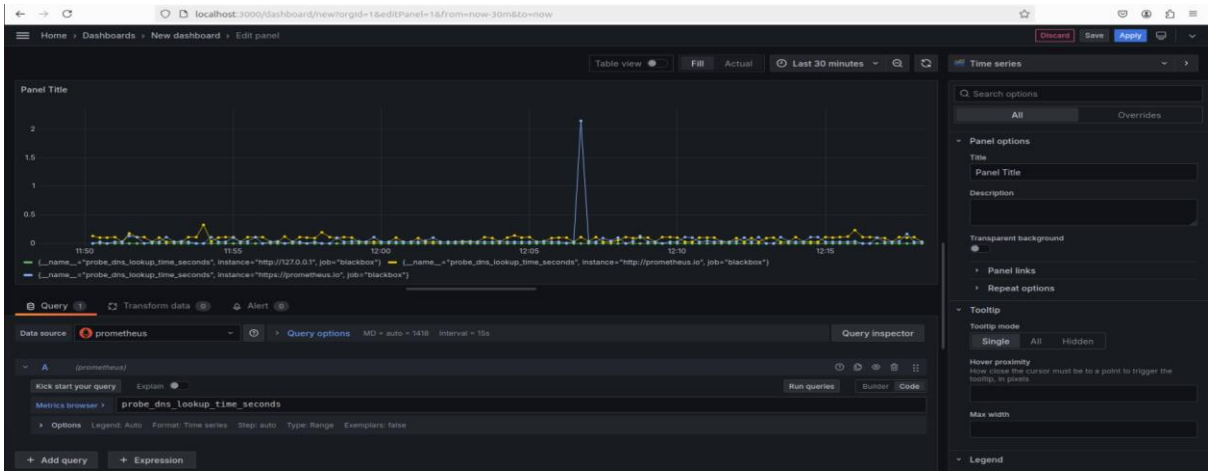
۴-۳-۴- Dns record :

خروجی در Prometheus:

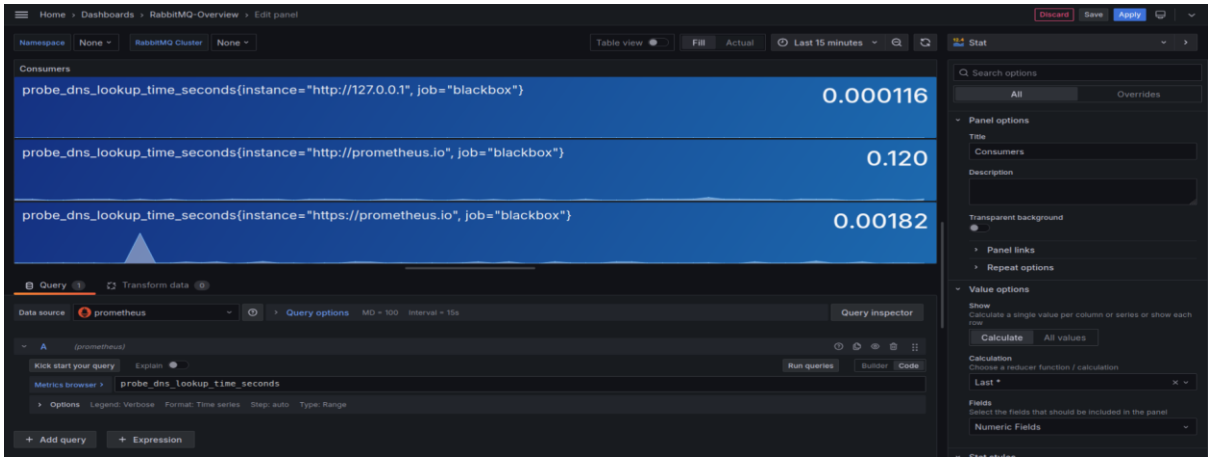


شکل ۴-۱۴۶: Dns record (Prometheus).

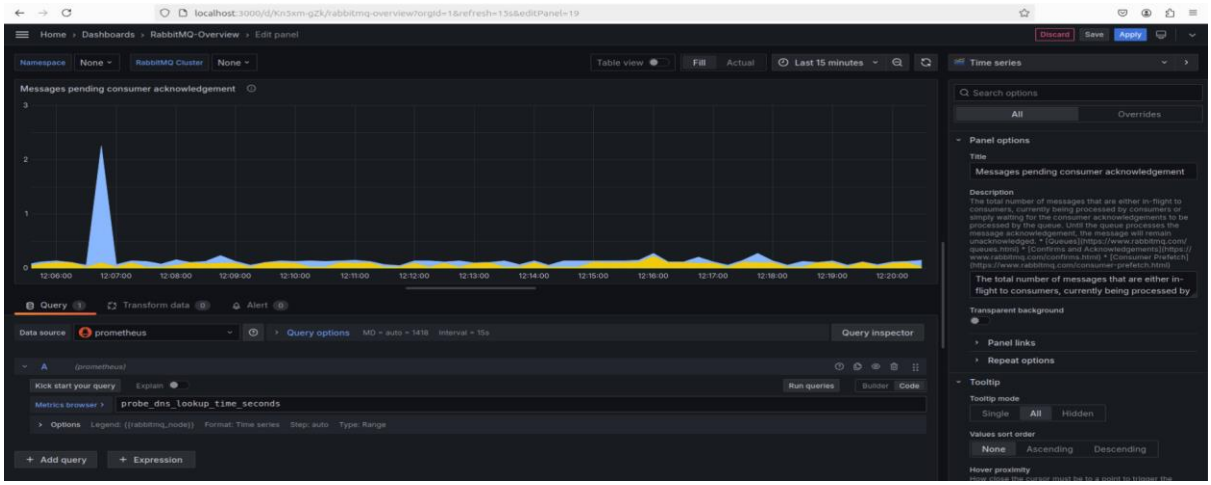
خروجی در Grafana:



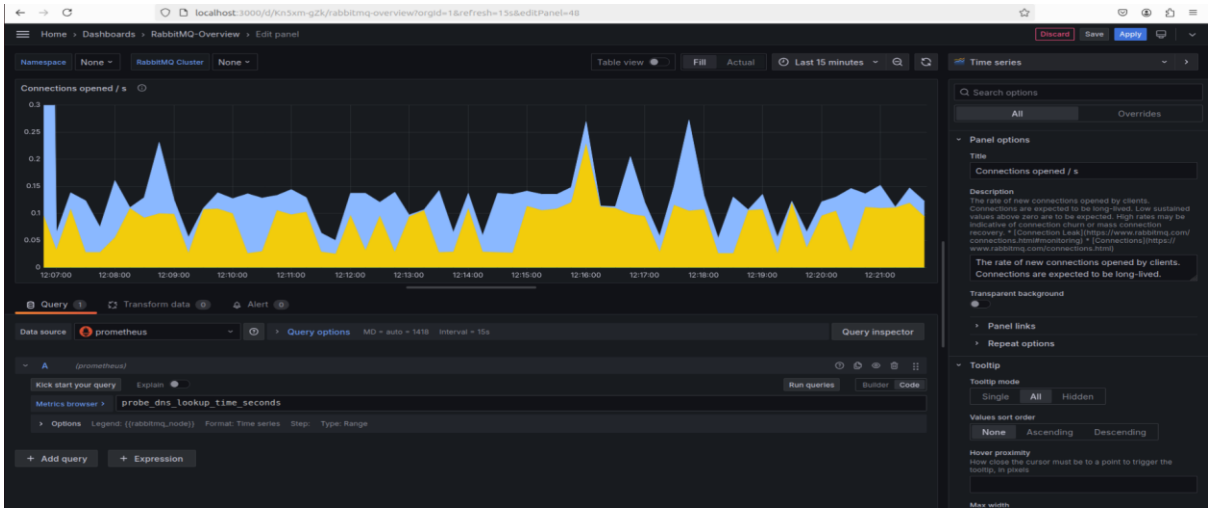
شکل ۴-۱۴۷: Dns record (Grafana).



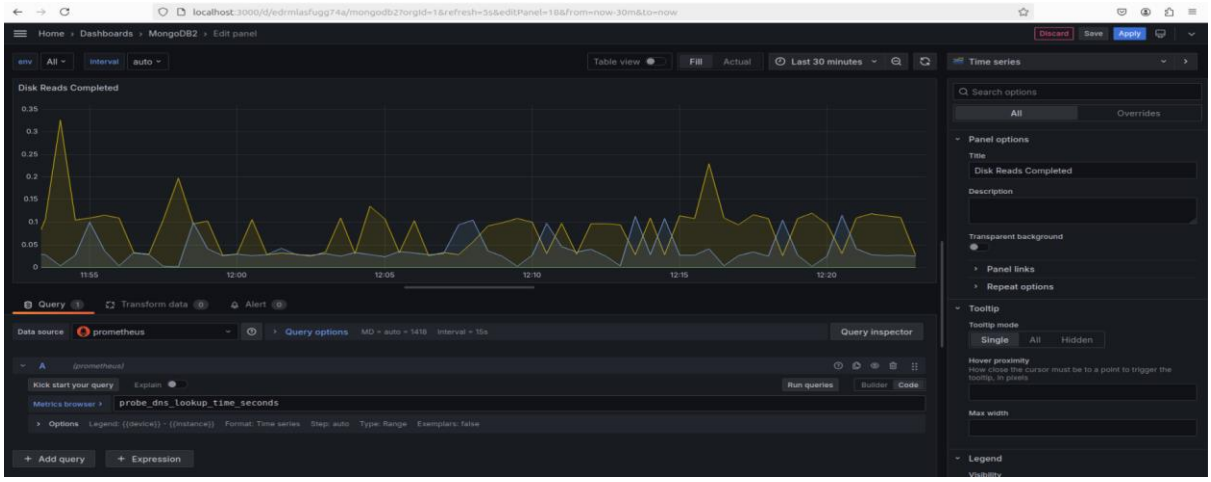
شکل ۴-۱۴۸: Dns record (Grafana).



شکل ۴-۱۴۹: Dns record (Grafana).



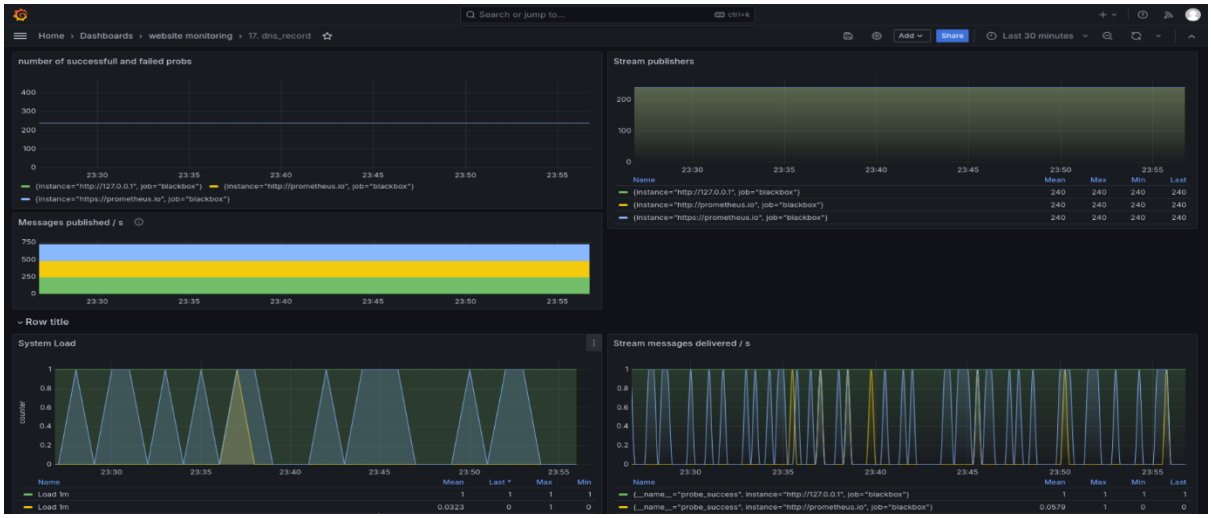
شکل ۴-۱۵۰: Dns record (Grafana).



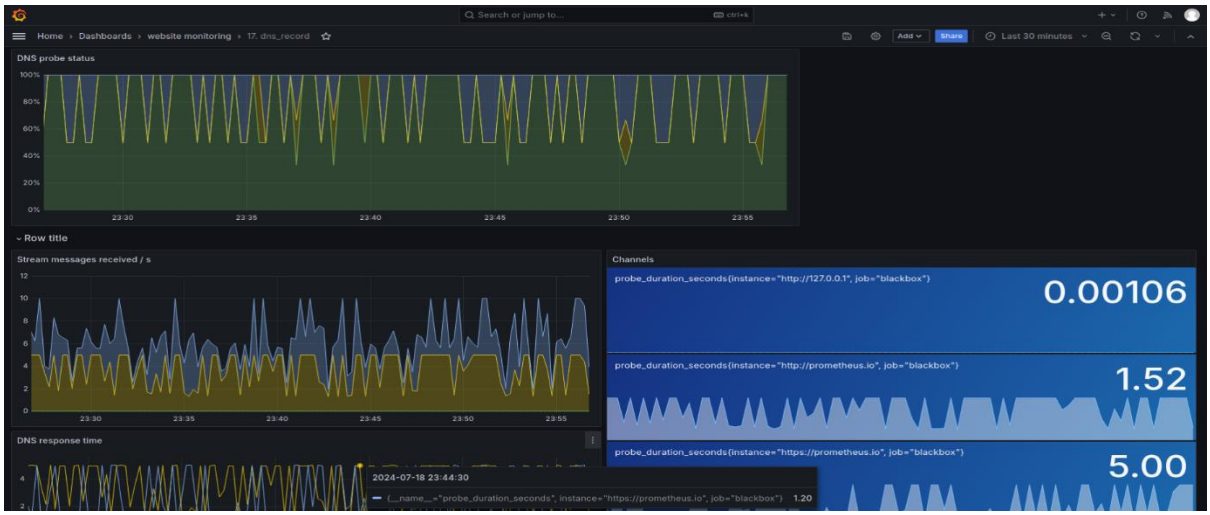
شکل ۴-۱۵۱: Dns record (Grafana).

خروجی نهایی داشبورد:

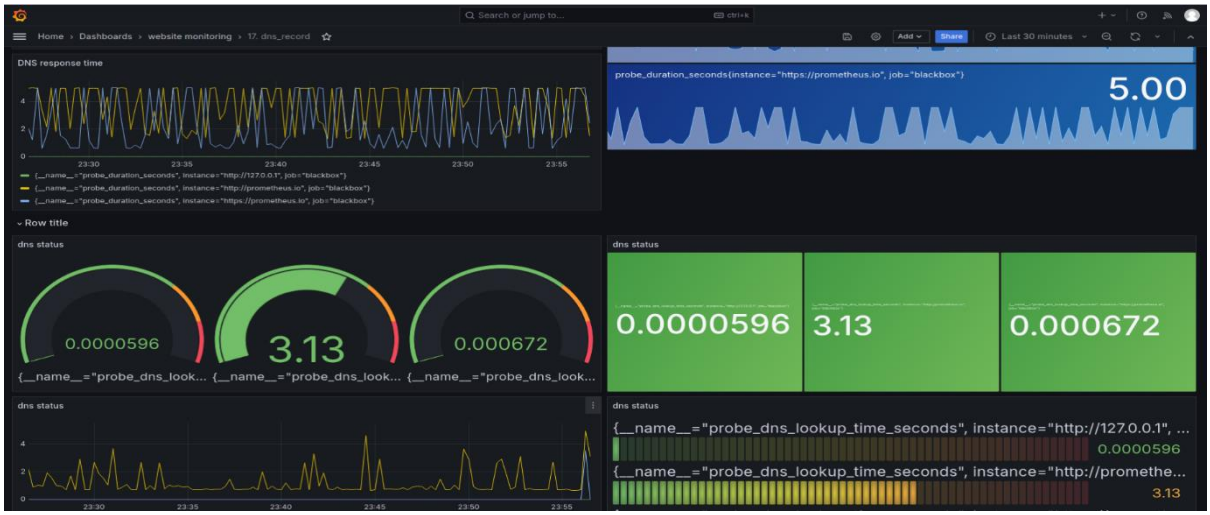
شکل ۱۵۲-۴: تصویر کامل داشبورد Dns record (Grafana).



شکل ۱۵۳-۴: تصویر کامل داشبورد Dns record (Grafana).



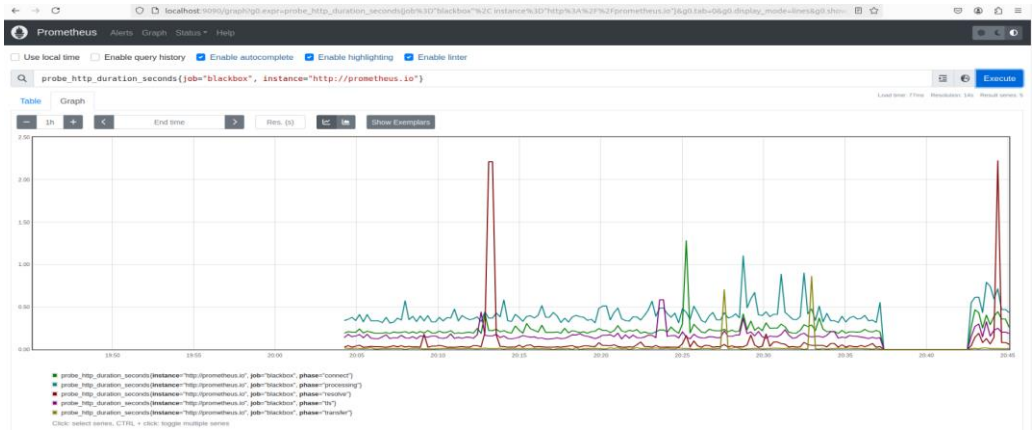
شکل ۱۵۴-۴: تصویر کامل داشبورد Dns record (Grafana).



شکل ۱۵۵-۴: تصویر کامل داشبورد Dns record (Grafana).

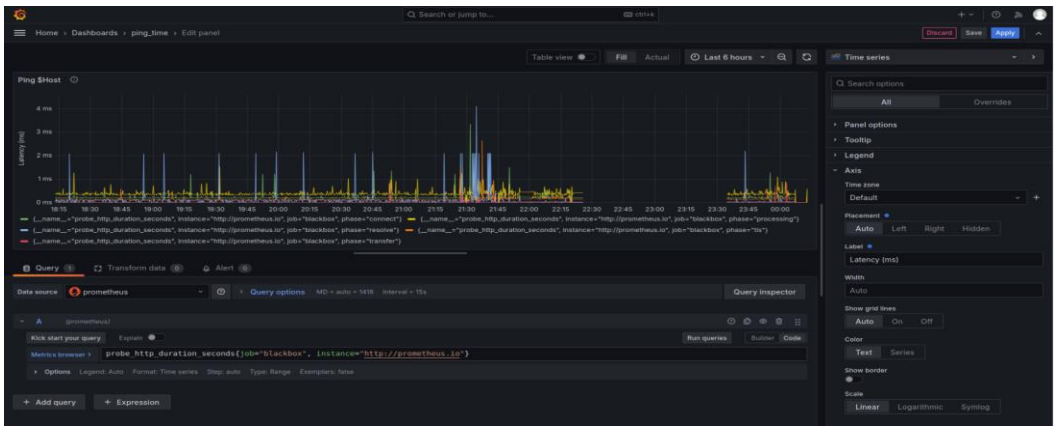
۵-۳-۴- Ping time :

خروجی در Prometheus:



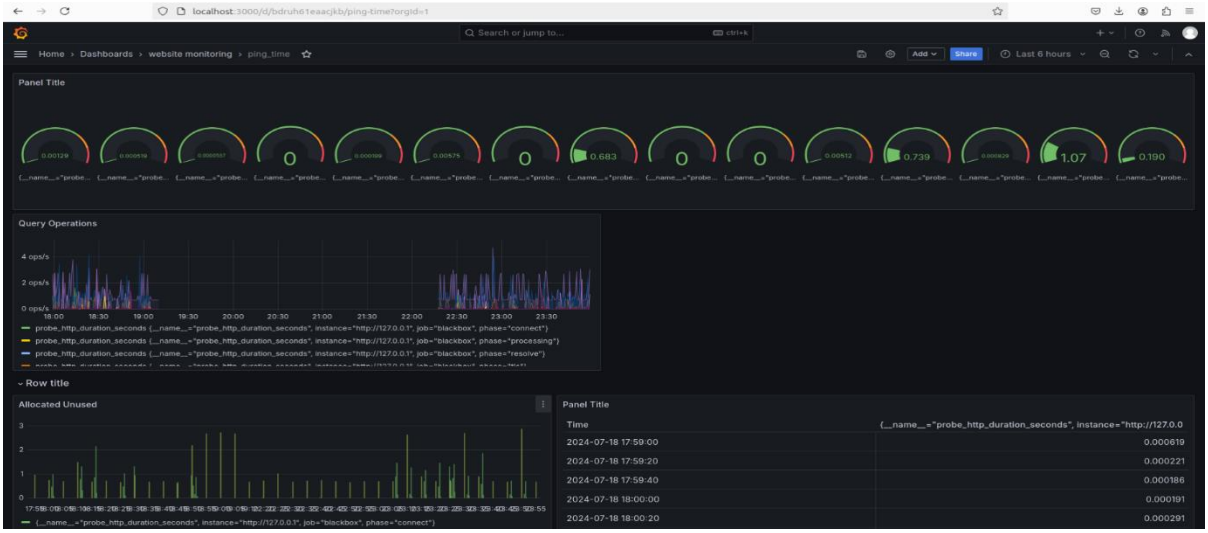
شکل ۱۵۶-۴: Ping time (Prometheus).

خروجی در Grafana:

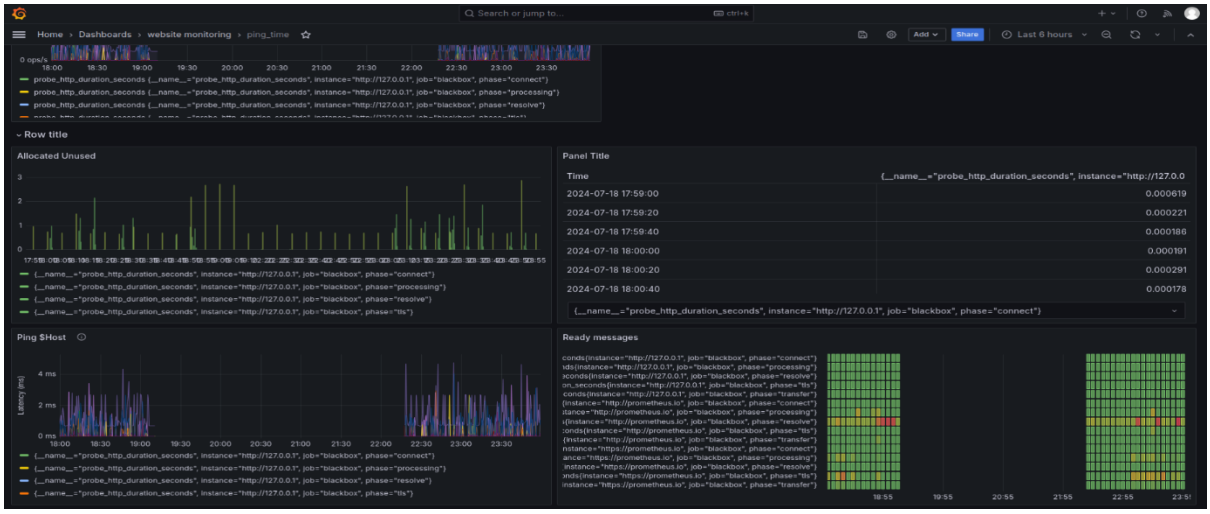


شکل ۱۵۷-۴: Ping time (Grafana).

خروجی نهایی داشبورد:



شکل ۱۵۸-۴: تصویر کامل داشبورد Ping time (Grafana).



شکل ۱۵۹-۴: تصویر کامل داشبورد Ping time (Grafana).

۶-۳-۴- وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی:

برای اجرای این قسمت دو راه وجود دارد:

۱-۶-۳-۴- **روش اول** ابتدا با استفاده از Influxdb یک دیتابیس ایجاد می کنیم:

```
amir@amir:~$ influx
Connected to http://localhost:8086 version 1.6.7~rc0
InfluxDB shell version: 1.6.7~rc0
> CREATE DATABASE influxdb_ssl;
> SHOW DATABASES;
name: databases
name
----
_internal
influxdb
influxdb_ssl
> exit
```

شکل ۱۶۰-۴: ایجاد دیتابیس با استفاده از influxdb.

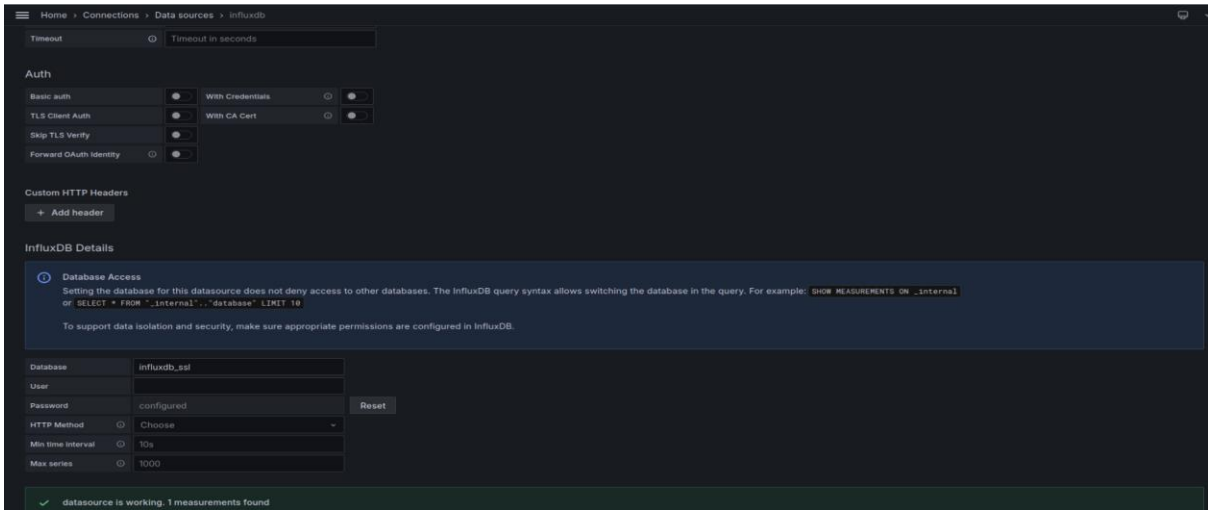
سپس با دستور Curl اطلاعات SSL سیستم را در این دیتا بیس ذخیره می کنیم:

```
amir@amir:~$ curl -i -XPOST 'http://localhost:8086/write?db=influxdb_ssl' --data-binary 'ssl_certificates
,host=example.com,san=example.com expiry=1633046400'
HTTP/1.1 204 No Content
Content-Type: application/json
Request-Id: a57a368e-42ba-11ef-8022-000000000000
X-Influxdb-Build: OSS
X-Influxdb-Version: 1.6.7~rc0
X-Request-Id: a57a368e-42ba-11ef-8022-000000000000
Date: Mon, 15 Jul 2024 14:58:07 GMT

amir@amir:~$ influx
Connected to http://localhost:8086 version 1.6.7~rc0
InfluxDB shell version: 1.6.7~rc0
> USE influxdb_ssl;
Using database influxdb_ssl
> SELECT * FROM ssl_certificates;
name: ssl_certificates
time                expiry     host        san
----                ------     ----        ---
1721054273249000033 1633046400 example.com example.com
1721055487937571394 1633046400 example.com example.com
> exit
```

شکل ۱۶۱-۴:ذخیره دستورات ssl با استفاده از دستور curl .

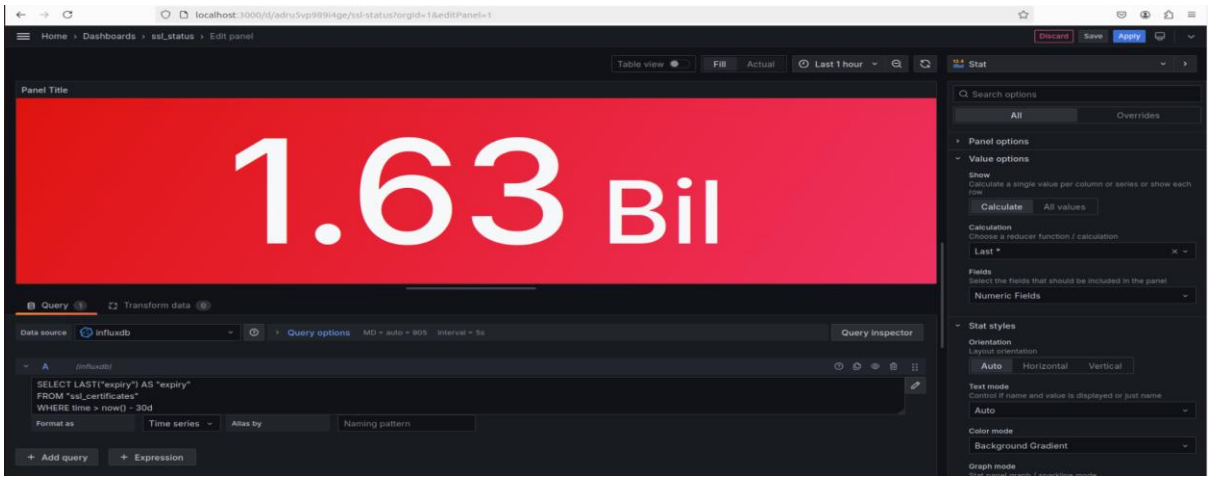
حال Influxdb را به گرافانا متصل و با استفاده از آن اطلاعات خیره شده را مانیتور می کنیم:



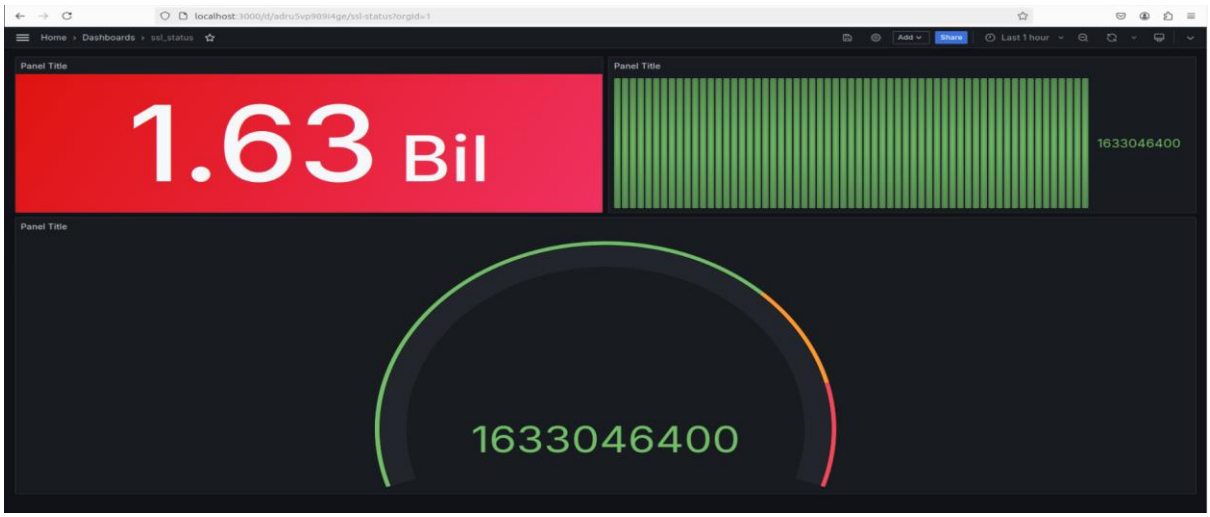
شکل ۱۶۲-۴: اتصال Influxdb به Grafana.

حال با تنظیمات جدید خروجی را در Grafana مشاهده می کنیم:

خروجی در Grafana :



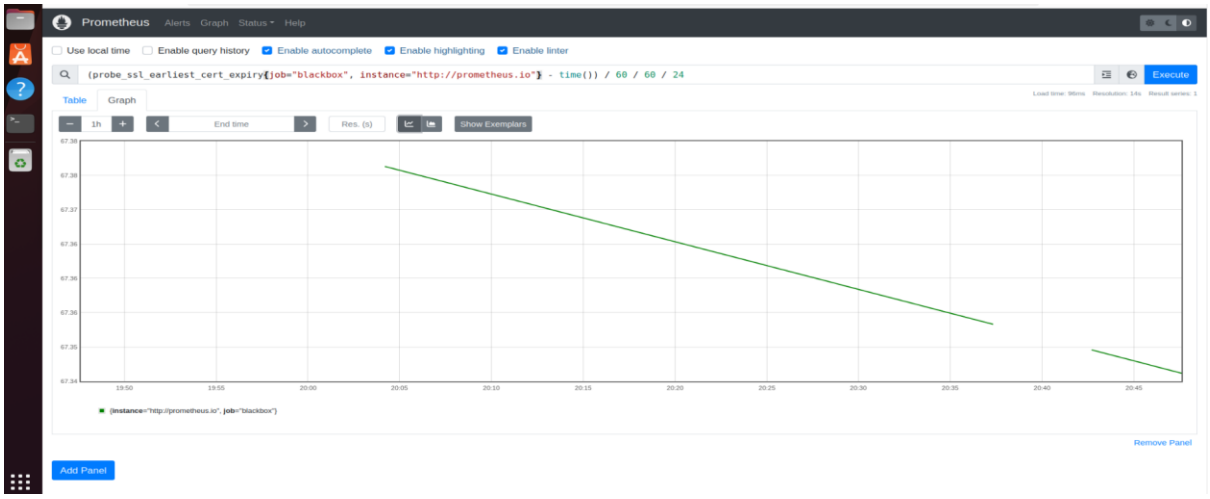
شکل ۱۶۳-۴: تعداد گواهی های SSl سیستم.



شکل ۱۶۴-۴: تعداد گواهی های SSl سیستم.

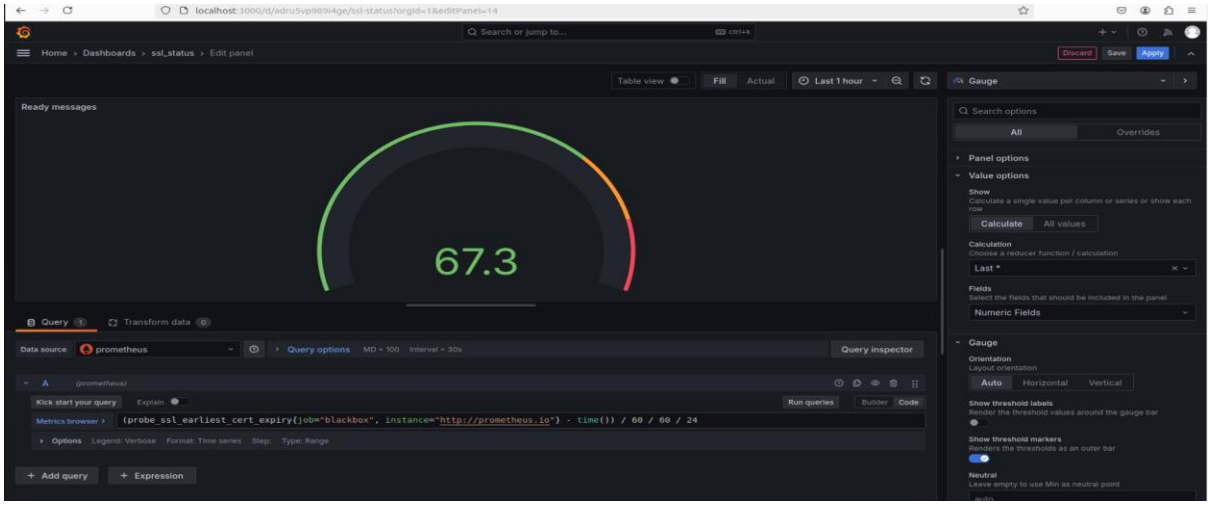
۲-۶-۳-۴- **روش دوم** با دستور و مشخص کردن تارگت در تنظیمات پرومتئوس:

خروجی در Prometheus:

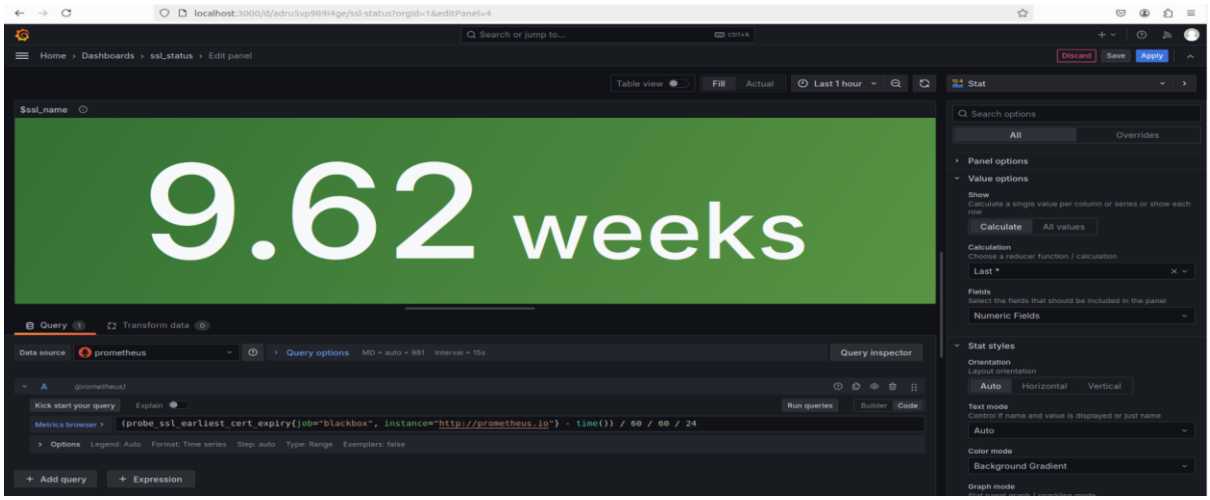


شکل ۱۶۵-۴: وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Prometheus).

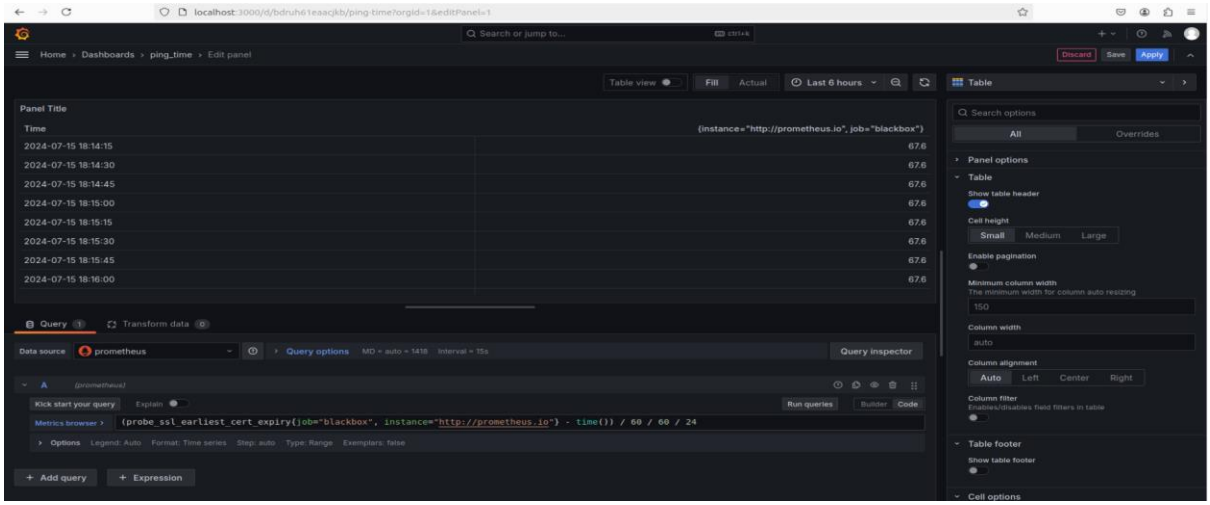
خروجی در Grafana:



شکل ۱۶۶-۴: وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Grafana).

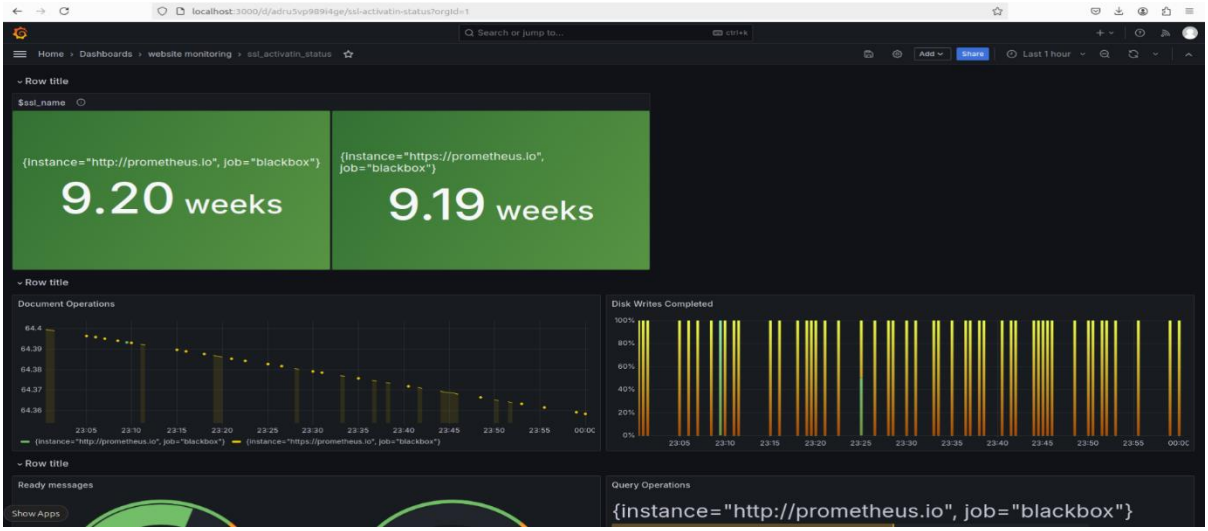


شکل ۱۶۷-۴: وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Grafana).

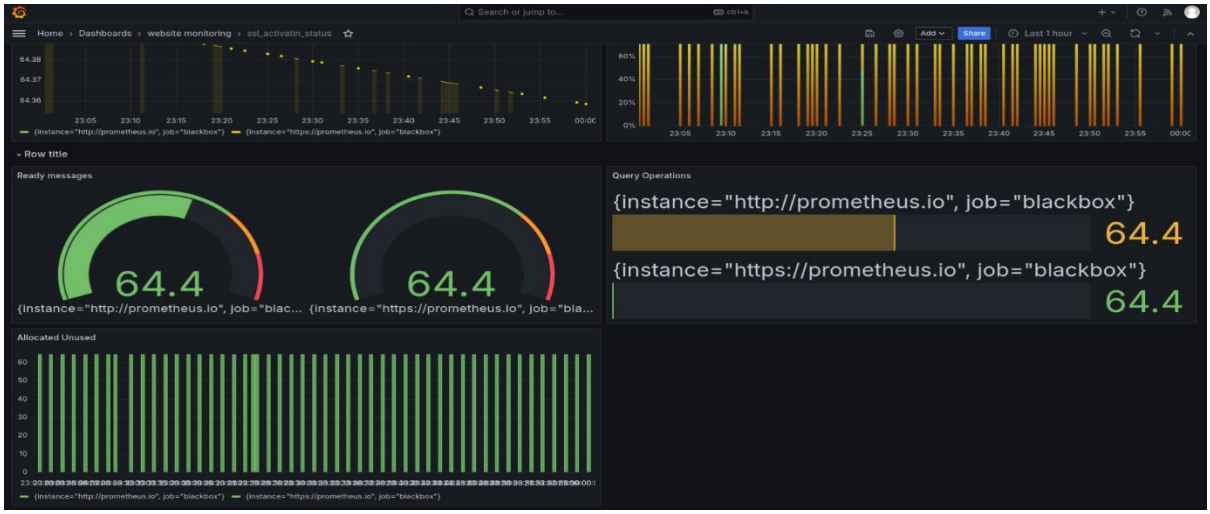


شکل ۱۶۸-۴: وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Grafana).

خروجی نهایی داشبورد:



شکل ۱۶۹-۴: تصویر کامل داشبورد وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Grafana).



شکل ۱۷۰-۴: تصویر کامل داشبورد وضعیت فعال بودن SSL مدت زمان گواهی و اطلاعات گواهی (Grafana).

فصل پنجم

نتیجه گیری و کارهای آینده

۱-۵- خلاصه، جمع‌بندی و نتیجه‌گیری

پروژه طراحی و پیاده‌سازی سامانه جامع مانیتورینگ، با تمرکز بر سه حوزه حیاتی شامل شبکه، سیستم‌عامل، و وب‌سایت‌ها، توانست تاثیر قابل‌توجهی بر بهبود مدیریت و بهره‌وری زیرساخت‌های فناوری اطلاعات بگذارد. این سامانه با استفاده از ابزارهای قدرتمند Prometheus برای جمع‌آوری داده‌ها و Grafana برای نمایش گرافیکی اطلاعات، بستری مطمئن برای پایش بلادرنگ و تحلیل داده‌های تاریخی فراهم آورده است.

در این سامانه، تمامی اطلاعات مرتبط با وضعیت منابع، عملکرد، و خطاهای احتمالی زیرساخت‌ها به‌صورت دقیق ثبت و تحلیل شده است. این ویژگی‌ها به مدیران و کارشناسان فناوری اطلاعات کمک می‌کند تا با شناسایی سریع مشکلات زیرساختی، مانع از بروز خرابی‌های بزرگ شوند. همچنین، امکان بررسی روندهای تاریخی، تحلیل الگوهای استفاده، و پیش‌بینی نیازهای آینده، این سامانه را به ابزاری کلیدی در مدیریت منابع فناوری تبدیل کرده است.

یکی از نقاط قوت برجسته سامانه، ارائه اطلاعات به‌صورت گرافیکی و بصری از طریق داشبوردهای قدرتمند Grafana است. این قابلیت باعث می‌شود که حتی مدیران غیرفنی نیز بتوانند وضعیت کلی زیرساخت‌ها را درک کرده و در تصمیم‌گیری‌های راهبردی مشارکت کنند. اطلاعات گرافیکی، نه تنها تحلیل داده‌ها را آسان‌تر کرده، بلکه به شناسایی روندها و نقاط ضعف احتمالی کمک شایانی کرده است.

این سامانه با ارائه هشدارهای بلادرنگ در مواجهه با مشکلات یا افت کارایی، نقش مهمی در کاهش زمان خرابی و بهینه‌سازی عملکرد منابع داشته است. کاهش هزینه‌های مرتبط با نگهداری و تعمیرات، افزایش سرعت واکنش به خطاها، و بهبود کارایی کلی زیرساخت‌ها از دیگر دستاوردهای کلیدی این پروژه به شمار می‌آیند.

با توجه به نیاز روزافزون سازمان‌ها به پایش مستمر زیرساخت‌های فناوری اطلاعات و اهمیت بالای واکنش سریع به مشکلات، سامانه جامع مانیتورینگ توانسته است به عنوان یک ابزار کاربردی و مؤثر در ارتقای بهره‌وری و مدیریت منابع شناخته شود. طراحی این سامانه نشان‌دهنده توانمندی‌های فناوری در بهبود فرآیندهای مدیریتی و ایجاد ارزش افزوده برای سازمان‌ها در عصر دیجیتال است.

۲-۵- پیشنهادات

- بهبود مقیاس‌پذیری سامانه: ارتقای قابلیت‌های سامانه برای مدیریت داده‌های حجیم‌تر و کاربران بیشتر.
- افزودن تحلیل‌های پیشرفته: استفاده از روش‌های تحلیل پیش‌بینانه برای شناسایی مشکلات احتمالی آینده.
- توسعه داشبوردهای قابل تنظیم: ارائه داشبوردهای منعطف‌تر برای سفارشی‌سازی بهتر بر اساس نیازهای مدیران.
- افزایش امنیت: تقویت مکانیزم‌های امنیتی برای حفاظت از داده‌ها و جلوگیری از دسترسی‌های غیرمجاز.
- آموزش تیم‌ها: برگزاری دوره‌های آموزشی جهت استفاده بهینه از سامانه و بهره‌برداری از قابلیت‌های آن.

مراجع

[۱] Grafana Labs. Grafana Overview. https://grafana.com/grafana/.

[۲] Prometheus (2024). Prometheus Documentation. Retrieved from https://prometheus.io/docs/

[۳] Prometheus Authors. Node Exporter Documentation. https://prometheus.io/docs/guides/node-exporter/.

[۴] curl Authors. curl: Command-line Tool for Transferring Data with URL Syntax. https://curl.se/docs/.

[۵] Prometheus Authors. Blackbox Exporter Documentation. https://prometheus.io/docs/guides/blackbox-exporter/.

[۶] Grafana Labs. Loki Overview. https://grafana.com/oss/loki/.

[۷] NGINX Authors. NGINX Documentation. https://nginx.org/en/docs/.

[۸] InfluxData. InfluxDB Documentation. https://docs.influxdata.com/influxdb/.

[۹] Promtail (2024). Promtail Documentation. Retrieved from https://grafana.com/docs/loki/latest/clients/promtail/

[۱۰] RFC 1123 (1989). Requirements for Internet Hosts -- Application and Support. Internet Engineering Task Force (IETF).

[11] SolarWinds Network Performance Monitor (NPM). https://www.solarwinds.com/network-performance-monitor

[12] ManageEngine OpManager. https://www.manageengine.com/network-monitoring/

[13] Zabbix. https://www.zabbix.com/

[14] Datadog. https://www.datadoghq.com/

[15] Monitis. https://monitis.com/

University of Tehran
College of Farabi
Faculty of Engineering
Department of Computer Engineering

Title

Design and Implementation of a Monitoring System for IT Infrastructure

By:

Amir Mirzaei Movahed

Under Supervision of:

Dr. Ahmad Alinejad

Consultant professor:

Dr. Amirhossein Kihanipour

A Project Report as a Requirement for the Degree of Bachelor of Science in Computer Engineering - Software

January 2025