Khai thác Prometheus tấn công Kubernetes

Le Minh Huong

VNU University of Engineering and Technology

Viettel Digital Talent Program Cloud - 2024

05/2024

Tóm tắt

Việc để lộ thông tin về Prometheus có thể khiến Kubernetes cluster dễ dàng bị tấn công. Chúng ta luôn nghĩ về việc dùng nhiều biện pháp phòng chống những sự tấn công đó, nhưng có một cách giải quyết đơn giản và cơ bản nhất: Che giấu thông tin của Prometheus. Những thông tin bị lộ có thể bao gồm nhiều dữ liệu như: nhà cung cấp cloud, các thành phần Kubernetes và thậm chí cả những dữ liệu bí mật. Người tấn công có thể sử dụng dữ liệu này và thực hiện các hành vi gây tổn hại đến hệ thống hay công ty, người quản lý hệ thống. Vì có thể thông qua Prometheus để tấn công vào Kubernetes, vậy nên khi sử dụng công cụ trên cần phải chú ý về mặt bảo mật, giảm thiểu việc bị tấn công.

Keywords: Prometheus, Kubernetes, Security

Mục lục

1. Mối liên hệ giữa Prometheus và Kubernetes

1.1. Kubernetes

Kubernetes, hay còn gọi tắt là k8s, là một nền tảng mã nguồn mở giúp tự động hóa việc quản lý, mở rộng và triển khai ứng dụng dạng container (Container Orchestration Engine). Nó giúp giảm phần lớn các công việc thủ công để triển khai và mở rộng các ứng dụng được container hóa.

Những đặc điểm chính của Kubernetes phải kể đến như là: Tự động hóa, khả năng mở rộng linh hoạt, dễ dàng quản lý tài nguyên, cho phép tích hợp với đa dạng dịch vụ, khả năng tự phục hồi vào tính bảo mật thông qua xác thực, ủy quyền, mã hóa dữ liệu.

Nhờ vào đó, các tổ chức, doanh nghiệp có thể dễ dàng mở rộng hệ thống mà không tốn quá nhiều thời gian, cải thiện độ tin cậy, tự phục hồi, hiệu quả hơn trong việc sử dụng tài nguyên. Kubernetes cũng cung cấp cho người dùng giao diện để có thể thuận tiện trong việc quản lý các clusters.

1.2. Prometheus

Prometheus là một dịch vụ mã nguồn mở, được sử dụng để theo dõi và cảnh báo về trạng thái hệ thống. Điểm nổi bật của Prometheus là khả năng thu thập, lưu trữ thông số và dữ liệu từ các dịch vụ theo khoảng thời gian đã được cài đặt sẵn. Hệ thống cũng cung cấp các API truy xuất kết quả, và đưa ra cảnh báo khi cần thiết. Prometheus còn sở hữu ngôn ngữ truy vấn PromQL, giúp giao tiếp hiệu quả với các dịch vụ monitor khác. Ngoài ra nó cũng cung cấp một giao diện người dùng để có thể theo dõi các dữ liệu, thống kê trên.

Prometheus có thể được mở rộng để hỗ trợ nhiều mục tiêu và workload. Vậy nên các công ty và doanh nghiệp có thể theo dõi hệ thống một cách hiệu quả. Cộng đồng người dùng và nhà phát triển lớn giúp cho dịch vụ luôn được cập nhật và phát triển nhanh chóng.

1.3. Mối liên hệ

Prometheus và Kubernetes là hai công nghệ containerization hàng đầu, đóng vai trò quan trọng trong việc quản lý và giám sát các ứng dụng container. Hai công cụ này lại thường được dùng để hỗ trợ lẫn nhau trong việc tối ưu hóa hiệu suất và độ tin cậy của hệ thống.

Prometheus giám sát hiệu suất ứng dụng container bằng cách thu thập dữ liệu về CPU, RAM, network… từ các container chạy trên Kubernetes. Những dữ liệu này có thể dùng để người dùng đánh giá hiệu suất ứng dụng, phát hiện và khắc phục sự cố khi phát sinh. Tình trạng các pod, node và thành phần khác trong cụm Kubernetes cũng được Prometheus theo dõi, giúp phát hiện sớm các vấn đề tiềm ẩn và tự khởi động lại các pod bị lỗi. Những dữ liệu trên cũng có thể được sử dụng để kích hoạt tính năng autoscaling trong Kubernetes, tự động điều chỉnh số lượng pod dựa trên nhu cầu tải. Prometheus khi được cấu hình sẽ tạo cảnh báo về các vấn đề hiệu suất hoặc tình trạng dịch vụ, phát sinh sự cố, từ đó người dùng có thể phản ứng kịp thời và ngăn chặn việc trở nên nghiêm trọng hơn

Tóm lại, Prometheus và Kubernetes là hai công cụ bổ sung cho nhau, giúp người dùng quản lý, giám sát và tối ưu hóa hiệu suất của các ứng dụng container một cách hiệu

quả. Việc sử dụng kết hợp hai công nghệ này giúp cho khả năng hiển thị của dịch vụ tốt hơn, hiệu quả cao, và có độ tin cậy ổn định, giảm thiểu thời gian chết.

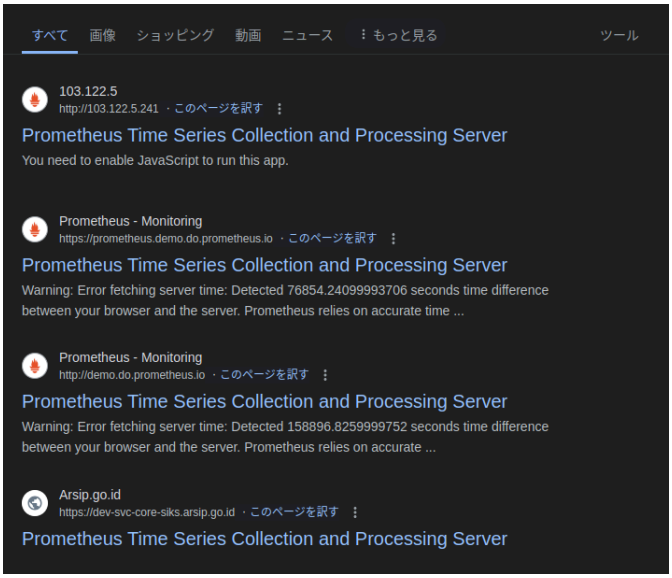
2. Khai thác Prometheus để tấn công Kubernetes

Chỉ cần sử dụng công cụ tìm kiếm Google, các Prometheus bị lộ có thể dễ dàng được tìm thấy. Prometheus lưu trữ dữ liệu theo dạng dòng thời gian (time series data), nghĩa là thông tin về các số liệu được lưu trữ cùng với thời gian ghi nhận. Bên cạnh đó, người dùng còn có thể sử dụng các cặp key-value (labels) để thêm ngữ cảnh cho số liệu. Mặc dù Prometheus khuyến khích sử dụng phương pháp xác thực cơ bản, nhưng tính năng này không được bật mặc định. Điều này đồng nghĩa với việc bất kỳ ai cũng có thể truy cập dữ liệu nếu Prometheus được mở trên mạng.

2.1. Vấn đề khi công khai các endpoint của Prometheus

Việc mở các endpoint của Prometheus sẽ cho phép bất kỳ ai cũng có thể thu thập dữ liệu: nhà cung cấp cloud, dữ liệu của các thành phần Kubernetes (pod, deployment, service…), thông tin về các container, cấu hình mạng, thậm chí cả dữ liệu bí mật (như token API).

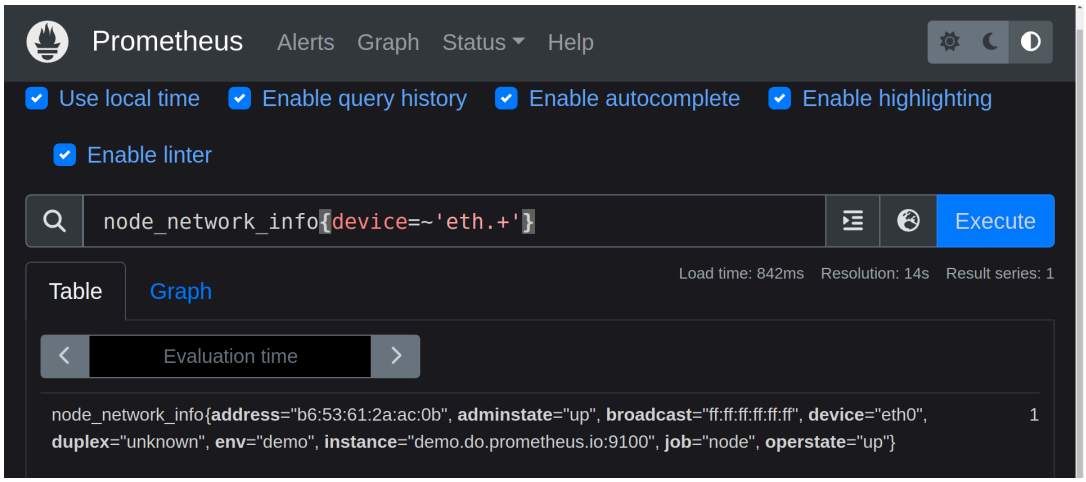
Từ đó gây ra các rủi ro về bảo mật, những kẻ tấn công có thể lợi dụng dữ liệu đó để xâm nhập vào hệ thống nhằm thực hiện các hành vi phạm pháp, gây tổn hại đến cá nhân hoặc doanh nghiệp. Mặc dù tiềm ẩn nhiều rủi ro, việc công khai các endpoint của Prometheus lại phổ biến.



2.2. Thực hiện tấn công

Để tấn công vào hệ thống, kẻ xâm nhập cần các dữ liệu, chỉ số về hệ thống đó. Thông tin này có thể dễ dàng trích xuất khi mở các Prometheus endpoints.

- Các dữ liệu về phần cứng có thể được tìm ra bằng cách sử dụng Node exporter. `node_dmi_info`, `node_network_info`, `kube_node_info`, thông qua các metrics này, kẻ tấn công có thể nắm được thông tin về nhà cung cấp mạng, mạng, địa chỉ ip, thông tin về os, ID của VPC (Virtual Private Cloud), hostname…



- Chỉ cần `ingress` và `services`, kẻ tấn công cũng có thể biết được đường dẫn của pod. Ingress controller trong Kubernetes hoạt động như reverse proxy và cho phép chuyển hướng các đường dẫn URL khác nhau đến các dịch vụ Kubernetes khác nhau. Trong khi đó, bộ cân bằng tải (load balancer services) được dùng để mở công khai các dịch vụ. Sử dụng các metrics `kube_ingress_path`, `kube_service_info` kết hợp cùng truy vấn PromSQL, sẽ xuất ra các thông tin về đường dẫn URL và các dịch vụ liên quan của ingress controller trong cluster hay thông tin của các services (load balancer services…). Như vậy, kẻ tấn công đã nắm được các thông tin: đường dẫn của URL (nhờ vào ingress) và các pod đang chạy (nhờ vào các service và nhãn của các pod).
- `kube_pod_info, kube_pod_container_info` được sử dụng để lấy được thông tin của các pod, và dữ liệu của các container trong pod (bao gồm dữ liệu về các dịch vụ như cơ sở dữ liệu, web server…)
- Kẻ tấn công cũng có thể tìm kiếm lỗ hổng bảo mật theo phiên bản của Kubernetes. `kubernetes_build_info` cung cấp thông tin về phiên bản một các đầy đủ (chính và phụ) của từng thành phần, gồm cả commit git và ngày chạy
- Trong một số phiên bản kubectl cũ hơn, metric `kube_secret_annotations` lưu trữ cấu hình được áp dụng gần đây nhất trong một annotation (bao gồm cả các secret). Điều này dẫn đến hậu quả là ngay cả khi secret chỉ có thể truy cập được bởi các service account và ràng buộc vai trò phù hợp, Prometheus vẫn có thể tiết lộ nội dung của secret.

Ngoài những metric trên, Prometheus vẫn có nhiều metric khác cho phép kẻ tấn công có thể lấy được thông tin về ứng dụng, dịch vụ, hay hệ thống, từ đó thực hiện các hành vi gây tổn hại hệ thống hoặc cá nhân, doanh nghiệp có liên quan. Và việc thực thi các metric trên có thể sẽ không được log lại. Prometheus có thể ghi log các câu truy vấn, nhưng tính năng này mặc định bị tắt. Mặc dù việc tắt ghi log truy vấn theo mặc định của Prometheus có thể giúp cải thiện hiệu suất, nhưng nó cũng có thể làm giảm khả năng theo dõi và phân tích hoạt động của người dùng trên máy chủ Prometheus. Để đảm bảo tính minh bạch và an toàn, người dùng cần cân nhắc việc lưu log theo mức độ chi tiết phù hợp.

2.3. Ảnh hưởng

Kẻ tấn công sau đó có thể sử dụng thông tin này để thực hiện nhiều loại tấn công.

- Khai thác tiền điện tử: Cài đặt phần mềm khai thác tiền điện tử vào các container trong Kubernetes cluster.
- Tấn công ransomware: Mã hóa dữ liệu trong cụm và yêu cầu tiền chuộc để giải mã.
- Lây nhiễm botnet: Thêm các node bị xâm nhập vào botnet để thực hiện các cuộc tấn công DDoS hoặc các hoạt động độc hại khác.

Lỗ hổng Prometheus bị lộ gây nhiều mối đe dọa nghiêm trọng cho cụm Kubernetes.

- Gây lộ, mất dữ liệu: Kẻ tấn công có thể truy cập và đánh cắp dữ liệu nhạy cảm được lưu trữ trong cụm.

- Gián đoạn dịch vụ: Việc vô hiệu hóa hoặc phá hủy các dịch vụ quan trọng trong cụm có thể gây gián đoạn hoạt động kinh doanh và phát triển của doanh nghiệp.
- Mất uy tín: Việc xâm nhập vào cụm Kubernetes có thể gây tổn hại đến uy tín của tổ chức.
- Chi phí tài chính: Để khắc phục hậu quả của một cuộc tấn công có thể tốn kém, bao gồm chi phí khôi phục dữ liệu, sửa chữa hệ thống và bồi thường cho khách hàng.

3. Biện pháp bảo vệ

Để bảo vệ hệ thống và dịch vụ, cũng như bản thân doanh nghiệp, việc cài đặt bảo mật cho nhà cung cấp đám mây, cluster, container và mã nguồn là cần thiết.

3.1. Bảo mật Prometheus

Bằng cách hạn chế quyền truy cập vào Prometheus, tức là chỉ cho những người dùng và hệ thống được cấu hình truy cập vào dịch vụ thông qua việc sử dụng xác thực và ủy quyền mạnh. Thêm vào đó cần giám sát những truy cập vào Prometheus để phát hiện hoạt động đáng ngờ. Trong trường hợp cần phải để lộ thông tin thì phải cấu hình Prometheus để chỉ các số liệu cần thiết được phơi bày.

3.2. Bảo mật Kubernetes

Ngoài việc sử dụng mạng riêng ảo (VPN) để bảo vệ lưu lượng truy cập giữa các node trong cụm, có thể dùng thêm Network Policy để kiểm soát lưu lượng truy cập giữa các pod. Cũng như bảo vệ cho Prometheus, cần hạn chế quyền truy cập vào các thành phần Kubernetes. Như đã đề cập trước đó, việc sử dụng kubectl bản cũ có thể gây lộ

thông tin quan trọng, vậy nên cần thường xuyên cập nhật Kubernetes có chứa bản vá lỗ hổng bảo mật. Image scanning cũng giúp cho việc phát hiện các phần mềm độc hại trong container.

3.2. Bảo mật mã nguồn

Việc bảo vệ hệ thống cũng phải được triển khai khi viết mã nguồn. Mã cần phải đảm bảo tính an toàn khi triển khai, tránh các lỗi phổ biến như SQL injection, XSS… Nên sử dụng các thư viện và frameworks được bảo mật tốt. Thường xuyên kiểm tra lại mã nguồn để phát hiện và sửa các lỗ hổng bảo mật.

Kết luận

Việc khai thác Prometheus là một mối đe dọa nghiêm trọng đối với các cụm Kubernetes. Bằng cách thực hiện các biện pháp bảo mật thích hợp, nguy cơ bị tấn công có thể được giảm thiểu và bảo vệ toàn vẹn dữ liệu.