خبرنامه برنا

Borna Network Managers
مدیران شبکه برنا

اخبار حوزه امنیت و شبکه (تیر ۱۴۰۴)

خبرنامه تخصصی «مدیران شبکه برنا»؛ نگاهی حرفه‌ای به دنیای امنیت شبکه هر ماه با ما همراه باشید تا در جریان جدیدترین تهدیدات سایبری، تحلیل رویدادها، به‌روزرسانی محصولات و راهکارهای نوین امنیت اطلاعات قرار بگیرید.

ویژه مدیران و کارشناسان شبکه و امنیت
(رایگان، تخصصی، به‌روز)

www.bornanetwork.co
info@bornanetwork.co
۰۲۱-۷۴۰۲۸

معرفی ریپوزیتوری Atmos

چارچوب Orchestration برای Terraform در معماری DevOps

ریپوزیتوری Atmos، ابزاری متن‌باز و مبتنی بر خط فرمان (CLI) است که توسط تیم Cloud Posse توسعه یافته و به منظور تسهیل در مدیریت زیرساخت‌های ابری طراحی شده است. این ابزار با هدف ایجاد ساختاری منسجم، قابل استفاده مجدد و سازگار با الگوهای GitOps، نقش بسزایی در بهینه‌سازی فرآیندهای مهندسی DevOps ایفا می‌کند.

قابلیت‌های کلیدی Atmos:

- **ادغام بومی با Terraform و Helmfile:** پشتیبانی مستقیم از ابزارهای استاندارد مدیریت زیرساخت جهت استقرار منابع در محیط‌های ابری متنوع.
- **پیکربندی‌های DRY و ماژولار با YAML:** پیاده‌سازی ساختارهای مبتنی بر Inheritance و Mixins جهت کاهش افزونگی و افزایش انسجام تنظیمات.
- **مدیریت Stacks و Components:** جداسازی منطقی و سازمان‌یافته اجزاء زیرساخت با قابلیت مقیاس‌پذیری بالا.
- **قابلیت اجرا در محیط‌های مختلف:** امکان استفاده از Atmos در محیط‌های محلی (Local) و خطوط CI/CD جهت استقرار خودکار.
- **سازگاری کامل با GitOps:** پیاده‌سازی فرآیندهای Infrastructure as Code بر مبنای ریپازیتوری‌های Git.

موارد کاربرد پیشنهادی:

- **مدیریت زیرساخت در معماری‌های چندسکویی (Multi-cloud)**
- **استانداردسازی فرآیندهای DevOps در سازمان‌های متوسط تا بزرگ**
- **کاهش پیچیدگی در کدنویسی و استقرار زیرساخت های ابری با حجم بالا**

مشاهده و دریافت ریپوزیتوری از GitHub:

این ابزار برای تیم‌های DevOps حرفه‌ای که به دنبال ساختاردهی پیشرفته، کنترل بیشتر و خودکارسازی پایدار هستند، یک انتخاب کلیدی محسوب می‌شود.

https://github.com/cloudposse/atmos

آسیب‌پذیری حیاتی CVE-2025-49719 در Apache Commons Text

یک آسیب‌پذیری جدی در کتابخانه Apache Commons Text شناسایی شده است که می‌تواند منجر به اجرای کد از راه دور (Remote Code Execution) شود.

شناسه: CVE-2025-49719

این آسیب‌پذیری ناشی از عدم بررسی کافی ورودی‌های کاربر در توابع Interpolator است، که به مهاجم اجازه می دهد با ارسال ورودی‌های دستکاری‌شده، کد دلخواه خود را روی سرور اجرا کند.

این آسیب‌پذیری به‌ویژه برای برنامه‌هایی که از Apache Commons Text برای پردازش ورودی‌های ناامن استفاده می‌کنند، خطرناک است. مهاجم می‌تواند با ارسال یک رشته خاص، کنترل کامل سیستم را به دست گیرد.

نسخه‌های آسیب‌پذیر: Apache Commons Text نسخه های 1.5 تا 1.9

توصیه‌ها:

1-به‌روزرسانی فوری: کتابخانه Apache Commons Text را به نسخه 1.10 یا بالاتر به‌روزرسانی کنید.

2-بررسی کد: تمام نقاطی که از Commons Text In-terpolator استفاده می‌کنند را بررسی و از ورودی‌های ایمن اطمینان حاصل کنید.

3-اعتبارسنجی ورودی: از اعتبارسنجی دقیق ورودی‌ها برای جلوگیری از حملات تزریق کد استفاده کنید.

منبع:

https://nvd.nist.gov/vuln/detail/CVE-2025-49719

حملات Fileless و تزریق در حافظه؛ دشمن مخفی EDRها

حملات مبتنی بر حافظه یکی از نقاط کور ابزارهای امنیتی سنتی محسوب می‌شوند. دلایل اصلی ضعف EDR در برابر این تکنیک‌ها عبارت‌اند از:

- **۱. تمرکز سنتی روی فایل و دیسک، نه حافظه**

بسیاری از EDRهای قدیمی‌تر طراحی شده‌اند تا فایل های روی دیسک را اسکن و تحلیل کنند.

بنابراین: رفتارهای مشکوکی که هیچ فایلی ایجاد نمی کنند (مانند بارگذاری Shellcode در حافظه) از دید آن‌ها پنهان می‌ماند، چون حمله‌ای در دیسک یا Log Event کلاسیک رخ نمی‌دهد، سیستم امنیتی اصلاً چیزی برای واکنش نمی‌بیند.

- **۲. Memory Scanning غیرفعال یا On-Demand**

قابلیت اسکن حافظه (Memory Scanning) در بسیاری از EDRها یا اصلاً فعال نیست یا فقط به‌صورت دستی (On-Demand) اجرا می‌شود.

در نتیجه: تهدیدی که در لحظه تزریق و اجرا می‌شود ممکن است هرگز شناسایی نشود حتی EDRهایی که این ویژگی را دارند، در حالت پیش‌فرض معمولاً آن را غیرفعال نگه می‌دارند (برای کاهش بار سیستم یا به دلیل خطاهای کاذب زیاد)

- **۳. استفاده از رمزنگاری، فشرده‌سازی و Obfuscation در Shellcodeها**

مهاجمان به‌راحتی از تکنیک‌های پیچیده‌سازی برای مخفی‌سازی کد مخرب خود استفاده می‌کنند:

Shellcodeها با الگوریتم‌های ساده (XOR، AES) رمزنگاری یا با encoding مثل Base64 پنهان می شوند. این Shellcodeها تنها در آخرین لحظه، در حافظه سیستم بازگشایی می‌شوند. بنابراین، تا قبل از اجرای واقعی، هیچ محتوای قابل تحلیل یا signature خاصی وجود ندارد که EDR آن را شناسایی کند.

- **۴. اجرای کد در فرایندهای قابل‌اعتماد (Trusted Processes)**

مهاجمان اغلب کد خود را داخل فرایندهایی تزریق می کنند که سیستم و EDR آن‌ها را به‌طور پیش‌فرض «اعتمادشده» می‌دانند، مثل:

- explorer.exe
- svchost.exe
- rundll32.exe
- wmiprvse.exe

از آن‌جایی که این فرایندها همیشه در سیستم فعال اند و نقش حیاتی دارند، EDR به‌سختی می‌تواند تفاوت بین فعالیت عادی و تزریق کد را تشخیص دهد؛ مگر با تحلیل رفتاری بسیار دقیق.

- **۵. ناتوانی در مانیتورینگ APIهای سطح پایین حافظه (Low-level Memory APIs)**

مهاجمان برای تزریق و اجرای کد در حافظه از توابع سیستمی سطح پایین استفاده می‌کنند؛ مثل:

- VirtualAllocEx
- NtWriteVirtualMemory
- CreateRemoteThread

بسیاری از EDRها: این APIها را به‌طور پیش‌فرض ثبت نمی‌کنند یا فقط در شرایط خاص (مانند فعال سازی Hunting Mode یا Custom Rules) آن‌ها را مانیتور می‌کنند.

بررسی تکنیک خطرناک BYOVD

سلاح مخفی مهاجمان برای دور زدن

اگر می‌خواهید فایل‌ها و فرآیندهای مشکوک را در محیطی امن و تعاملی تحلیل کنید، ابزار ANY.RUN را امتحان کنید!

ANY.RUN چیست؟

یک ابزار sandboxing پیشرفته و تعاملی است که به متخصصان امنیت سایبری اجازه می‌دهد بدافزارها و فعالیت‌های مشکوک را در زمان واقعی (Real-Time) بررسی کنند. این ابزار با ارائه محیط مجازی قابل تنظیم، به شما امکان می‌دهد فرآیندهای ناشناخته را بدون خطر برای سیستم واقعی خود، به‌صورت زنده تحلیل کنید.

ویژگی‌های کلیدی:

-تحلیل Real-Time برای مشاهده رفتار بدافزارها.

-پشتیبانی از انواع فایل‌ها و اسکریپت‌های مخرب.

-امکان تنظیم محیط‌های متنوع (مثل سیستم‌عامل‌ها و تنظیمات دسترسی).

-گزارش‌دهی دقیق از فعالیت‌های شبکه و سیستم.