云安全的下半场：原生安全*

刘文懋
绿盟科技

关键词：云原生　原生安全

云安全将成为纯安全问题

关于云安全的未来，著名咨询机构高德纳（Gartner）有两个比较有意思的论断：

论断 1：网络安全的未来在云中 [1]

随着云计算的日益普及，企业上云已经成为必然的趋势。Gartner 曾做出一个预测：在 2020 年前，50% 的企业将业务工作流放到本地需要作为异常事件进行审批，公司“无云”的策略会和现在“无网络”的策略一样少。可见，云计算将成为企业各项应用必不可少的服务平台和基础设施，那么讨论网络安全怎么做，就必须要考虑面向云计算的网络安全怎么做，例如虚拟网络隔离、东西向的入侵检测，等等。

论断 2：云安全会变成单纯的安全 [2]

云计算与各行各业的 IT 基础设施进一步融合，云或是基础，或为组件。例如，5G、边缘计算和工业互联网，都需要云计算技术构建云化的基础设施或编排平台，那么这些新型系统的基础设施安全，本质上就是云计算 IaaS/PaaS/CaaS 的安全；此外，如欺骗技术、靶场技术等新的网络安全机制，或多或少地使用了虚拟化、容器等技术，因而，这些与云计算技术融合后，就形成了新的、普适的安全技术，即“just security”。

一方面，云化的基础设施和平台需要安全防护，用传统安全手段赋能云计算；另一方面，云计算的各种新技术、新理念（如软件定义、虚拟化、容器、编排和微服务等），也在深刻变革着当前的安全技术发展路线。因而，未来的云安全，一定会将“云”这个定语去除，云安全等价于安全本身，即安全技术必然覆盖云计算场景，安全技术必然利用云计算技术。

云计算的下半场：云原生计算

如很多其他新技术一样，云计算起源于美国，但千万不能照搬美国的云计算发展过程到国内，复制一套相似的产品。事实上，在云计算的上半场，即从云计算诞生至今，中美两国走了两条不同的发展路线，这与两国的各自国情是有密切关系的。

具体而言，美国的云计算发展路线是先 SaaS（Software as a Service，软件即服务）后 IaaS（Infrastructure as a Service，基础设施即服务）。SaaS 是最早的云计算服务形态。早在 1999 年，甲骨文（Oracle）前执行官马克·贝尼奥夫（Marc Benioff）就创办了 Salesforce，这是当前最大的客户关系管理（CRM）SaaS 服务提供商。经过 20 年的发展，美国的 SaaS 服务已经深入企业业务，平均每个企业会用到 1427 个云服务，平均每名员工会用到 36 个云服

*本文是基于 2020 年绿盟技术嘉年华上的同名主题演讲做的内容扩充。

[1] 详见 https://www.valtix.com/uploads/secure-access-service-edge-gartner.pdf。

[2] 详见 https://www.slideshare.net/TimmothySilva/webinar-gartner-top-security-trends-for-midsize-enterprises。

务[3]。SaaS 的安全防护主要是以云端接入安全代理（CASB）为主，因而国外的 CASB 市场巨大，但其挑战在于需要适配大量 SaaS 服务，所以这个市场的玩家目前主要是 Skyhigh、Netskope 等巨头。

近几年来，随着企业进一步将业务云化，特别是将 IT 基础设施替换为 IaaS 服务中的虚拟计算资源，通过软件定义广域网（SDWAN）连接分支结构、云端资源，形成全栈云化、全分支机构云化的趋势。此时，虽然 IaaS 整体营收还远不及 SaaS，但其增长率激增，2019 年的公有 IaaS 服务增长率达到了 37.3%[1]，远超云服务的总体增长率（17.5%）。如亚马逊云服务（AWS）这样的公有 IaaS，其安全防护主要是利用亚马逊（Amazon）提供的各类接口，在虚拟网络、虚拟机层面提供网络和终端防护，Gartner 把虚拟机层面的安全防护技术称为云工作负载保护平台（CWPP）。

中国的云计算发展是从虚拟化起步，从私有云到公有行业云，走出了一条具有中国特色的发展路线。里程碑是开源的 IaaS 项目 OpenStack 在国内兴起，国内厂商（如华为、华三、EasyStack 等企业）基于 OpenStack 研发了各自的云平台，此时国内的云计算需求主要是将硬件服务器虚拟化，再加入多租户管理、网络隔离等需求，因而，多数云计算服务商提供的是私有云的解决方案。通常商用私有云系统是封闭的，缺乏对网络流量按需控制的应用接口，因而，针对这类私有云的安全机制多为安全资源池，通过路由、VLAN 或开放网络接口将流量牵引到资源池进行处理。随着节约成本、集约化管理和提供增值服务等需求的进一步增强，具有云平台开发能力的服务商基于前述的私有云平台，提供了公有 IaaS 的服务。然而，这种公有 IaaS 服务与 AWS、阿里云不太一样，它们具有鲜明的行业特性。例如，为政府提供的政务云，将所有政府下属机构的服务器迁移到新的云平台上，提供政务相关的服务。这样的公有 IaaS 服务，本质上还是前述的 OpenStack 系的系统，其封装了自服务功能，并提供行业相关的合规服务和增值服务，因而其安全防护技术也可以基于安全资源池之上，提供面向租户的安全即服务（Security as a Service）。

但总体而言，这样的上云实践只是“形”上的改变，还远没有到“神”上的变化。过去两年的行业发展表明，无论是中国还是美国，云计算的新增长点已经都转向云原生相关的领域，如容器即服务（Containers as a Service，CaaS）、编排技术、微服务、DevOps 等，至此云计算进入下半场，其驱动力无外乎以下两方面：

1. 应用快速交付和开发运营一体化。DevOps 的开发运营模式已经深入人心，由开发团队驱动的容器化部署、应用编排等，事实上提出了新型的云交付模式。

2. 新型 IT 基础设施部署。如 5G、工业互联网和边缘计算场景下，资源受限，有资源虚拟化等需求，大量使用了容器、编排和微服务等技术，也使得云原生应用未来可期。

云原生相关的技术栈在过去 3~5 年中得到了快速发展，以 Docker、Kubernetes、Istio 为代表的容器运行时、编排系统、服务网格已经成为事实上的标准，而 API 网关、无服务框架也在快速演进中。可以预计，未来 5 年内，云原生相关的技术会在互联网企业、金融、运营商等行业得到大量应用。笔者认为，云原生就是云计算的下半场，谁赢得云原生的赛道，谁才真正赢得了云计算。

原生安全：基于云原生、无处不在的安全

如果说云安全的未来等价于纯安全，而云计算的下半场是云原生，那不妨也做个推论：云原生的未来也会等价于原生安全。那么，什么样的安全才是原生安全呢？笔者认为原生安全有两个特点：基于云原生，无处不在。也就是说，使用了云原生的技术，能适用于各类场景。

[3] 数据来自 Gartner Security & Risk Management 2019 峰会。

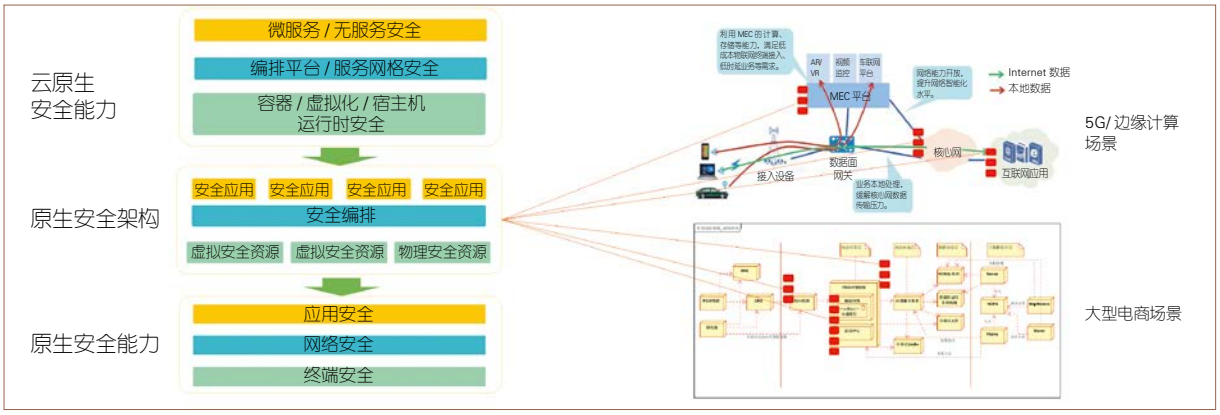

图 1　原生安全的演进

原生安全的发展会有三个阶段，如图 1 所示：

1. 安全赋能于云原生体系，**构建云原生的安全能力**。当前云原生技术发展迅速，但相应的安全防护匮乏，就连最基础的镜像安全、安全基线都不尽如人意。因而应该研究如何将现有成熟的安全能力（如隔离、访问控制、入侵检测、应用安全）应用于云原生环境，构建安全的云原生系统。

2. 云原生的新特性具有诸多优点，例如轻快不变的基础设施、弹性的服务编排、开发运营一体化等。因而，安全厂商会开始研究如何将这些能力赋能于传统安全产品，通过软件定义安全的架构，**构建原生安全架构**，从而提供弹性、按需、云原生的安全能力，提高“防护－检测－响应”闭环的效率。

3. 当安全设备或平台云原生化后，就能提供（云）**原生的安全能力**，不仅适用于通用云原生场景、5G、边缘计算等场景，甚至可以独立部署在大型电商等需要轻量级、高弹性的传统场景，最终实现无处不在的安全。

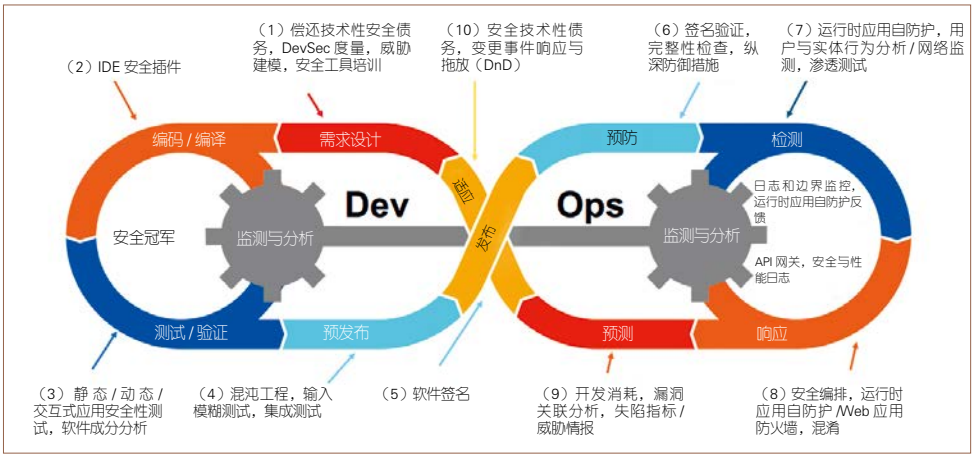

图 2　DevOps 闭环

安全左移与右移

如果考虑云原生应用的生命周期，则应关注 DevOps 的整个闭环（见图 2），即从开发、编译、持续集成 / 持续部署（CI/CD），到运行时运营。由于容器的生命周期极其短暂，对于攻守双方来说，在短期内都无法应用现有的武器库或安全机制。所以在云原生安全的初期，攻击者会关注代码、第三方库和镜像这些生命周期长的资产，而防守者也应该关注安全编码、开源软件脆弱性管理、镜像和仓库脆弱性评估，以及安全基线核查。这些安全机制基本上处于 DevOps 的左边的闭环，因而我们将这些安全举措称为**安全左移**，以区别于传统在运行时做的安全运营工作。事实上，过去两年，国内外很大一部

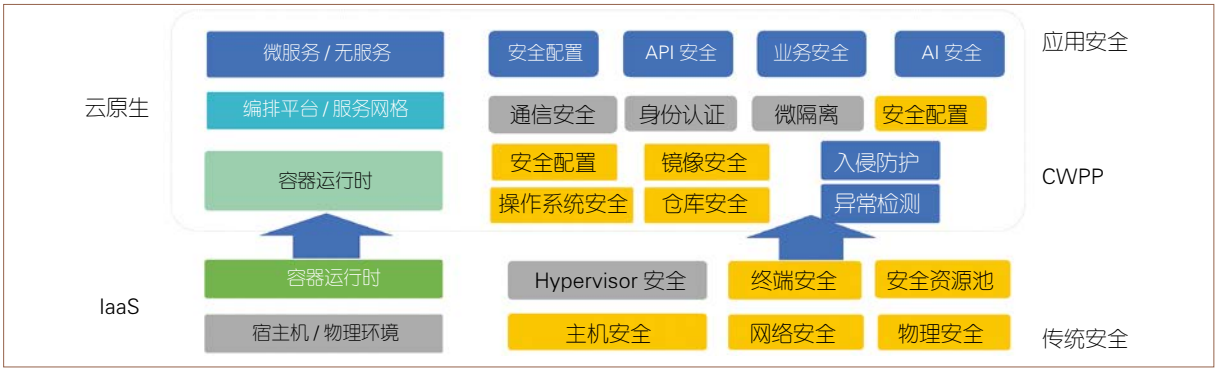

图 3　云原生安全技术栈

分容器安全解决方案都聚焦在这部分内容。

当然，攻防永远是成本和收益之间的平衡，如果防守方能做好对长生命周期资产的持续风险评估和脆弱性缓解，那么攻击者的成本显然会升高，他们下一步就会借助自动化的攻击手段，尝试在运行时攻击微服务、无服务和容器，进而借助短暂存在的容器横向移动寻找其他可持久化的资源。此时，容器工作载荷的行为分析、容器网络的入侵检测、服务网格的 API 安全和业务安全，则又会成为防守者的新重点，我们将这种重点放在运行时的安全思路称为**安全右移**。

	msg	event_type	process	user	cpu	process_len	container_name
0	原进程出现了新用户	01-02	bash	abc	0	4	k8s_data-dispatcher_data-dispatcher-deployment...
12	新进程启动，进程名长度过长异常	02-04	/usr/bin/python_abc	root	0	10	k8s_action-dispatcher_action-dispatcher-deploy...
7	原进程出现了新路径	01-03	/abc/bin/sh	root	0	2	k8s_zookeeper_zookeeper-1_default_9998eacf-85d...
11	原进程CPU偏高	01-01	/pause	root	100	5	k8s_POD_data-dispatcher-deployment-84d9f7f59d-...

图 4　偏移行为基线的可疑行为

无论是安全左移还是安全右移，其实都是考虑到云原生环境中的脆弱性、面临的威胁和风险，在有限的安全投入前提下，做出当前最有利的安全方案。

总体而言，云原生的（安全）技术栈如图 3 所示，可见，云原生安全不只是独立的容器、编排或微服务，而应该完整地考虑整个云计算系统的所有组件及其安全功能需求。

运行时安全

从技术实现来看，运行时安全比开发安全更难。本节重点讨论如何实现运行时安全，主要分为异常行为检测、安全防护和 API/ 业务安全三方面。

异常行为检测

攻击者对云原生系统的攻击，可以分为已知威胁和未知威胁。对容器而言，最危险的已知威胁莫过于容器逃逸，文献 [2, 3] 讨论了从文件链接、容器运行时、宿主机操作系统内核等多个层面实现的容器逃逸。对于这类攻击，从内核层面捕获进程、网络等各种行为，通过规则即可及时发现潜在的逃逸行为。

而对未知的攻击而言，最好的办法就是为容器的正常行为建立基线。通常运行微服务的容器只运行少数进程，且行为可预测，所以建立的基线能够很好地刻画相关容器的行为，进而可以在运行时及时发现偏离基线的可疑行为（见图 4）。

安全防护

运行时安全防护可分为面向容器的安全防护和面向微服务 / 服务网格的安全防护。在容器层面，需要重点关注的是容器组建的网络中是否存在入侵行为，例如针对容器及其打开服务的侦查、横向移

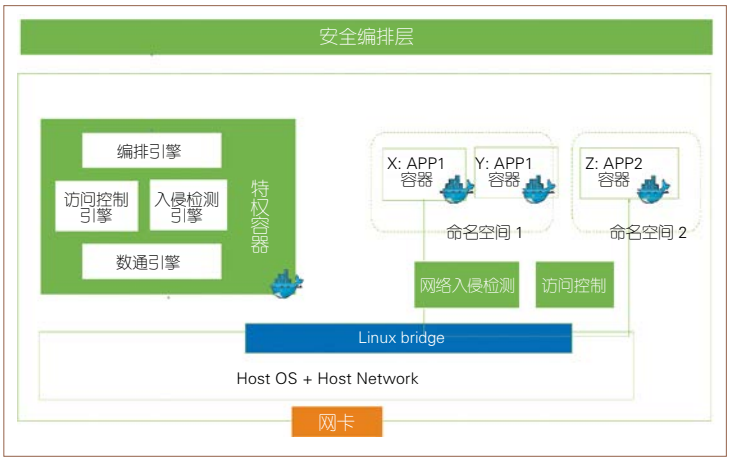

图 5　基于特权容器的容器安全防护

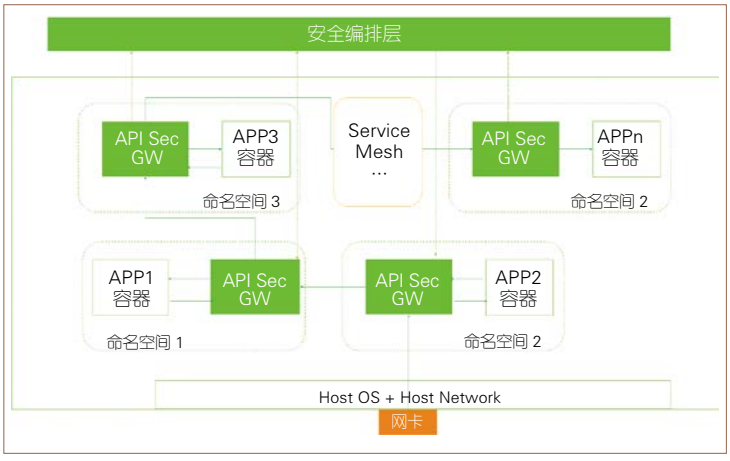

图 6　基于 Sidecar 的微服务安全防护

动，此时通常可使用特权容器的方式，部署网络入侵检测组件，即可实现微隔离、访问控制、网络流量监控、入侵检测和防护等功能，如图 5 所示。

在微服务和服务网格层面，则需要考虑将安全能力部署到服务网格中的每个微服务最近侧，因而使用边车（Sidecar）的方式，将安全微网关部署在应用容器旁，如图 6 所示，这样可以实现应用的认证授权、通信加密等应用层面的安全防护等功能。

API 和业务安全

业务安全是离客户最近且价值最大的安全功能，然而云原生场景非常复杂，很难有统一的业务安全模型；此外，在微服务和服务网格的场景下，服务之间大部分是通过 API 调用实现，这与当前 Web 应用存在大量人机交互的情况完全不同，因而 API 安全在云原生场景下也存在很大的差异。

从技术上看，要实现 API 和业务安全，首先是获得 API 调用的可观察能力（visibility），当前有很多开源项目具有这样的能力，但其在开发、构建过程中的侵入性各有不同，如表 1 所示。一般而言，侵入性越强，其最后获得的 API 请求信息越多，但在既有开发流程和项目中的部署也越难。

其中，Jaeger 通过在代码中插桩的方式，能够获得所有调用顺序和参数，因而从理论上就能建立非常精准的 API 调用参数和序列的基线；而 Sidecar 使用反向代理的方式，无法获得调用序列，只能通过分析，启发式地获得近似基线，其上限可以逼近 Jaeger 所得到的基线。至于具体采用哪种观测方法，则取决于客户侧的部署情况。

表 1　开源项目的可观察能力对比

	Zipkin	Jaeger	Skywalking	Sidecar
代码侵入性	是	是	否	否
镜像侵入性	是	是	是	否
带 trace ID	是	是	是	否
带请求参数	是	是	否	是
支持语言	C#/Go/Java/JavaScript/Ruby/Scala/PHP	Go/Java/Node/Python/C++	Java/Node/PHP/Go	全部
可在 K8S 上部署	是	是	是	是
是否支持云原生环境部署	是	是	是	是

图 7　KubeEdge 安全防护原型系统

原生安全：未来的安全

我们在前面谈到，如果云原生安全成为了原生安全，那就说明云原生已经融入到了各行各业，成为普适的云计算场景。事实上，国家已经开始大力推动新基建战略，包括 5G、物联网、工业互联网等信息基础设施，云计算、人工智能等新技术基础设施，数据中心等计算基础设施等。而这些基础设施，未来或多或少都会与云原生技术有所联系。

例如在边缘计算的场景下，目前行业中主流的开源边缘计算平台，如 OpenNess、KubeEdge 和 StarlingX 均采用了容器和编排技术，并且可以提供第三方微服务。我们将前面所提到的云原生安全技术栈移植到这三个边缘计算系统中，证明了该方案是完全可行的。图 7 展示了面向 KubeEdge 的安全防护原型系统。

而在 5G 核心网（5GC）中，我们发现在切片技术应用中，资源层会使用虚拟化和容器技术，如一些开源的 5GC 项目也在使用容器编排技术。事实上，一些主流的商用 5GC 网元也采用了容器技术；而控制层则会通过 RESTful/HTTP 协议进行通信。

可见 5GC 的网元自身防护，可以使用前述云原生安全技术栈进行加固和防护；而网元的业务侧安全，则可以借助前述 API/ 业务安全的基线方式刻画正常网元业务，进而发现可疑的网元请求。

当新基建推动大量云化基础设施采用了云原生的技术路线，当云原生的安全能力可以部署在云化或非云化的环境中，那我们真的就可以说，未来的安全就是“原生安全”。 ■

刘文懋

CCF 高级会员、理事。绿盟科技创新中心总监，星云实验室负责人。主要研究方向为云计算安全、网络安全。

liuwenmao@nsfocus.com

参考文献

[1] Donna Goodison. Gartner: IaaS Public Cloud Services Market Grew 37.3% In 2019[OL].(2020-08-10).https://www.crn.com/news/cloud/gartner-iaas-public-cloud-services-market-grew-37-3-in-2019.

[2] 【云原生攻防研究】容器逃逸技术概览 .[OL].(2020-02-20).https://cloud.tencent.com/developer/article/1590363.

[3] 未能幸免！安全容器也存在逃逸风险 [OL].(2020-09-25).https://www.freebuf.com/articles/250918.html.