基于网络模式识别和匹配的容器安全防御方法与装置

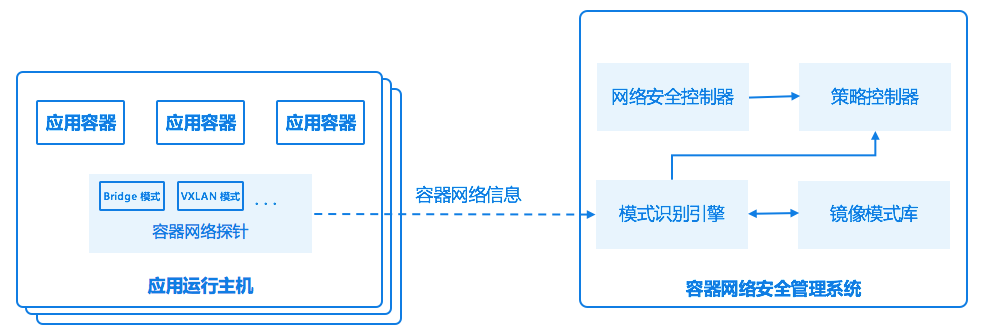
# 背景

网络安全已经成为企业安全的重中之中，而绝大多数企业的网络安全都仅仅关注在网络边缘加固上，比如通过网络物理隔离等方式把企业IT核心基础设施和互联网进行完全隔离。但是，仅仅关注边缘网络加固是不足以满足企业网络安全要求的，甚至会让企业放松在内部网络的管控上。很多时候，一旦边缘网络加固被破坏的话，企业内部缺乏安全保护的系统和数据就完全暴露出来。而且，边缘网络加固仅仅防御了企业外部网络，但是企业内部网络同样面临严峻的安全威胁，一旦企业内部网络中的某台主机被黑客所控制，那么很容易对内网的其他主机和系统产生安全威胁。

而在企业内部网络安全方面，针对应用／主机IP和端口进行网络防火墙控制是绝大多数企业采用的方式。但是随着企业应用上云的不断深化，这种静态配置的方式已经难以为继。尤其在PaaS场景下，应用的部署架构是动态的，PaaS平台会根据应用和平台节点的运行状态对应用的部署架构进行动态调整，比如在应用负载过大时自动增加应用实例，或者当平台的某个节点不健康时自动把应用迁移到其他节点上。这种情况下，以往针对应用IP进行防火墙设置的做法，就不能满足动态需求了。而容器技术的大量引入又提出了更高的要求，因为每个代表应用运行实例的容器的IP可能只是一个应用平台内部有效的IP地址，对平台之外的企业网络来讲，并不能实际访问。同时，容器在运行时可以根据使用场景采用不同的容器网络模式，进一步增加了对容器化应用进行网络监控的难度。

# 系统描述

本方案提出了一种网络安全管控方法和系统，能够自动识别并适配容器采用的运行时网络模式，有针对性的对容器的网络行为数据进行收集和分析，同时通过容器镜像信息来标识和关联应用，并基于策略对应用的网络行为进行威胁行为甄别和干预。本方案的架构图如下所示：

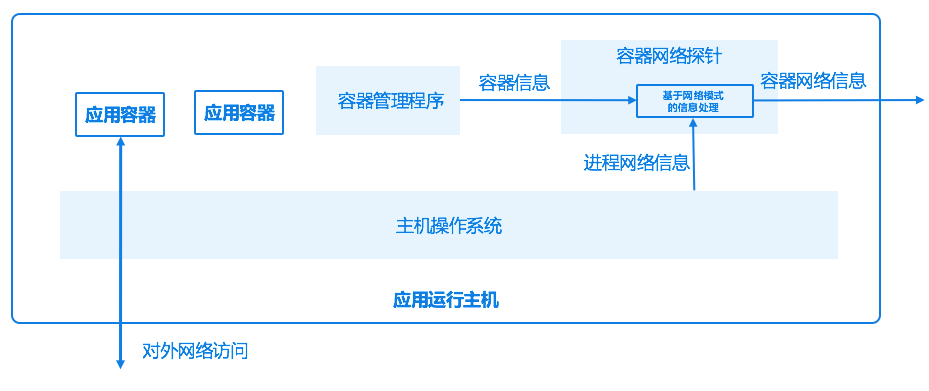


## 信息收集

通过在每个运行应用容器的主机上部署并运行“容器网络探针”程序，该程序实时收集主机上所有进程的网络访问信息，包括网络访问流量、网络主动对外访问和被访问的IP和端口信息等；同时，该程序会通过查询容器管理程序，把容器和进程进行关联，从而形成基于容器的网络访问数据。由于容器在运行时可以指定不同的网络模式，而在不同的网络模式下容器网络数据包承载的信息差别巨大。鉴于此，“容器网络探针”程序会自动从容器管理程序获取所有容器的运行时网络模式，并根据容器网络模式采取对应的网络数据包处理方式对容器网络数据包进行解析，比如：

* 对于“网桥”网络模式：网络数据包的IP源地址和端口已经被替换为主机的IP和端口，因此需要根据网桥映射关系，使用原始容器IP和端口记录容器网络信息；
* 对于“VxLAN”网络模式：原始容器网络数据包被主机使用VxLAN的封装格式进行包装后发送，因此需要对VxLAN网络包进行拆包，使用原始容器网络数据包中的元信息记录容器网络信息；
* 对于“Host”网络模式：容器网络数据包的元信息即为实际信息，直接使用数据包中的信息记录容器网络信息即可。

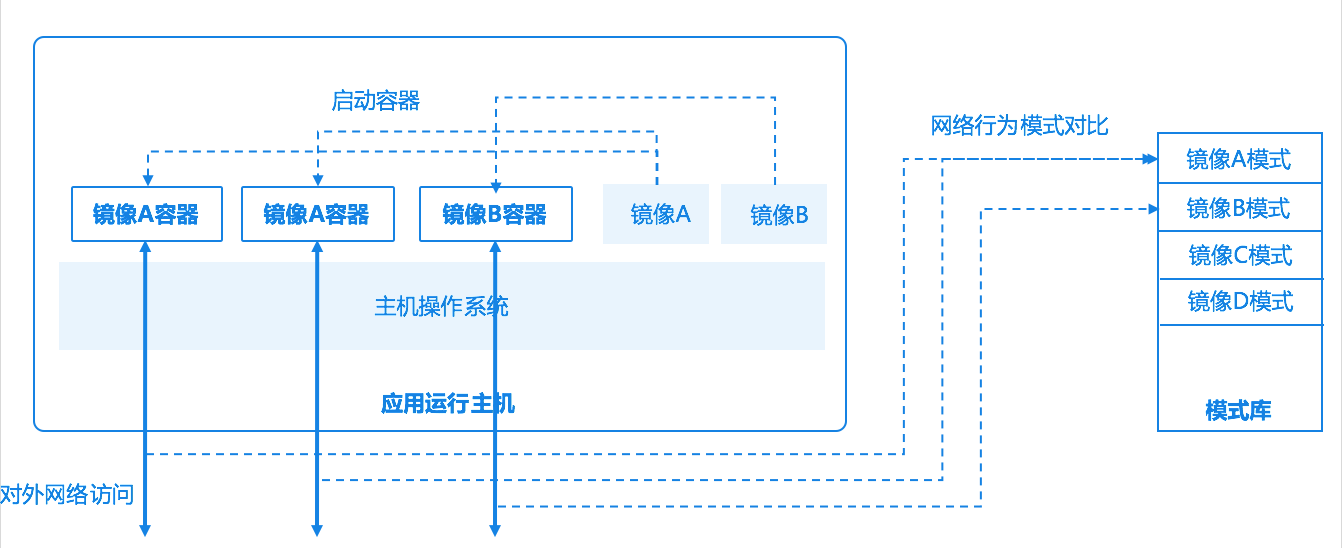
同时，“容器网络探针”还采用可扩展的方式支持更多的容器网络模式。最终经过解析处理过的网络信息包括但不限于容器IP及端口信息、容器的镜像信息、容器主动对外访问及被访问信息、网络流量等。上述架构如下所示：



请注意，虽然上图只涉及到了单个主机，但是在多主机的环境中也是类似的机制，每个主机都保持了上述类似的部署架构。

## 监控目标识别

容器技术是通过一种层次结构的文件系统将应用和环境封装到了一起，而封装的产物就是镜像。镜像是容器化应用的静态表示和数据存储格式，当镜像中封装的程序被一个由容器技术管理的进程运行起来时就成为一个容器。容器是由于容器化应用都是通过镜像发布的，同一个镜像派生出的容器的访问模式应该是类似的。因此，本系统会根据容器镜像对应用进行分类和标记，并以镜像为单位进行网络行为模式识别和比对，即当分析到一个应用容器的网络行为时，会通过匹配该容器对应镜像的网络模式来进行风险识别。



如上图所示，从同一镜像启动的容器的网络行为会和同一镜像标示的网络模式进行关联。请注意，虽然上图只涉及到了单个主机，但是在多主机的环境中也是类似的机制。

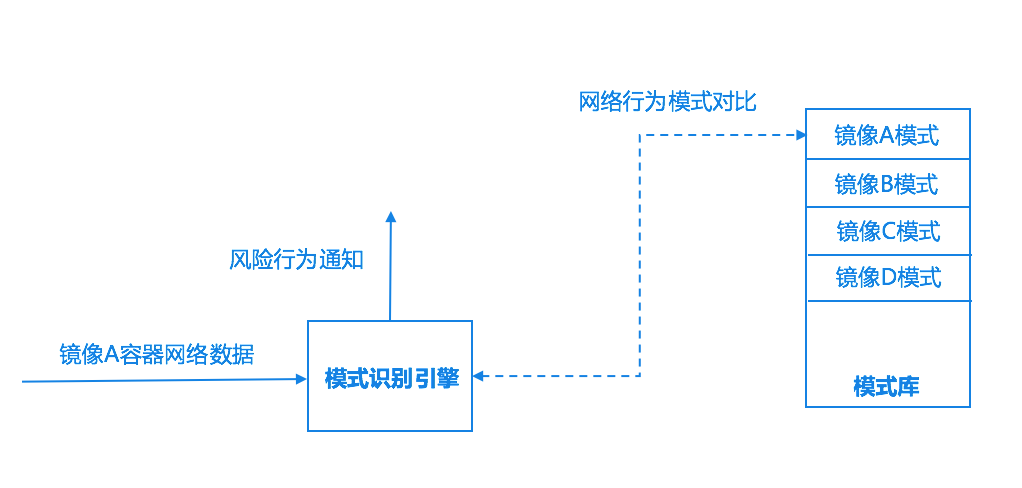
## 模式识别

当“模式识别引擎”接收到容器网络数据后，会根据该容器的镜像是否被识别过而展开不同的行为。如果该数据对应的容器镜像从来没有识别过，则把该镜像标示为“待识别镜像”；如果该数据对应的容器镜像已经完成模式识别，则把该镜像标示为“已识别镜像”。

“模式识别引擎”对于收集到的应用网络数据，会提取数据的特征信息，比如目标地址、带宽、数据量等，进行模式识别计算，比如分类分析、聚类分析等。通过不断积累同一镜像的网络数据进行模式识别，并最终产生该镜像的模式定义。

## 风险识别

当“模式识别引擎”接收到容器网络数据后，会判断是否该镜像是“已识别镜像”，如果是，则根据该容器的镜像匹配相应的网络模式。如果发现收集的网络数据不匹配相应的镜像网络模式，则识别该网络行为为一个风险行为。架构如下图所示：



# 系统工作流程

1. 所有主机上“容器网络探针”程序从系统和容器管理程序收集容器网络行为数据，通过匹配相应的容器网络方式对数据进行分析处理，并最终把处理后的信息实时发送给本方案中的“模式识别引擎”，进行模式匹配分析。
2. 网络安全管理员通过“网络安全控制器”设置模式学习方式，可以设置为“自动识别”或者“手动识别”，自动学习即一旦发现某一网络数据来自未知镜像则自动进行识别，而手动识别则仅识别指定的镜像。同时，可以设置学习过程定义，比如设置针对某一个镜像的学习时间或者数据量。下述流程针对“自动识别”，手动识别过程类似处理。
3. 如果“模式识别引擎”发现采集到的网络数据来源于一个未被识别过的容器镜像，“模式识别引擎”将标示该应用镜像为“待识别镜像”，并根据预先设定的学习阶段定义，比如设定的固定时间内，对该镜像的所有容器的网络访问模式进行学习，形成应用网络访问模式并存入“镜像模式库”，并标示该应用镜像为“已识别镜像“
4. 如果“模式识别引擎”发现采集到的容器信息来源于一个“已识别镜像“，则和”镜像模式库“中该镜像的网络访问模式进行匹配，如果发现不匹配的情况则通知”策略控制器“，后者根据预设置的策略执行相应动作，包括但不限于生成威胁告警、阻止容器的可疑网络访问、停止可疑容器的运行等；
5. 可以通过本方案中的“网络安全控制器“对镜像的网络访问模式进行配置管理；
6. 可以通过本方案中的“网络安全控制器“对安全管控策略进行配置管理；

# 创新点

* **特别针对容器化运行的应用进行网络监控**，能够自动关联进程的网络行为到某应用容器上，在信息实时收集和分析的基础上，自动对容器网络威胁进行干预；
* **以容器镜像为网络模式识别对象**，极大的降低了模式识别的目标范围，并适应当前越来越广泛的容器化应用部署场景；
* **能够自动识别容器的网络模式并对网络行为数据进行相应的分析处理**，比如对于采用封装网络模式的容器，探针能够捕捉到封装网络包内部的原始容器网络信息；
* **系统内置的镜像模式库中可以预制大量常见的镜像网络模式**，加快企业安全建设进程；