Network filtering in a distributed app

Brian Cheung and Sam Wang

ABSTRACT

We will examine the vulnerabilities of a distributed application without using any layers of security. Then we will implement Pod Security Policies and Data Plane Filters to imporve the security of the application itself and filter out suspicious network traffic. The filtering of certain packets based on information such as the source/destination IP addresses and protocols can further reduce the attack surface and strengthen current security practices for applications. Through our research, we hope that the implementation of filtering out network traffic will lead to an increase in security performance.

1. INTRODUCTION

There exists many vulnerabilities within networks and distributed applications, and the implementation of network filtering as well as strong pod security policies can help mitigate these threats.

Figuring out how to effectively utilize network filtering and pod security policies to work together will hopefully result in a strong security infrastructure that can successfully defend against all attacks and will be adopted by all applications.

We will be researching how network filtering and pod security policies function and how they can be applied to improving the security of an application. By understanding how each work, we will be able to understand how to effectively filter out network traffic that seems suspicious, thus reducing the amount of potential attacks from the start, and we will understand how to write a concrete and complete pod security policy to defend against the attacks that slip through the filter.

Pod Security Policies (PSP) provides a framework that will layout the rules of how a pod can operate and ensures that they run with the correct privileges and access. Furthermore, PSPs are used so that those operating Kubernetes clusters can control pod creations and limit what pods can access. When a pod is deployed, the PSP acts as a gate-keeper that will compare the pod security configuration to what is defined in the PSP. Some examples of how PSPs can limit pod behaviors include, preventing privileged pods from starting and controlling its privilege, restricting pods from accessing host namespaces, filesystem, and networks, restrict the amount of user/groups that a pod can run, and more. Network filtering is the practice of monitoring the inflow/outflow of packets in a network. There are two types of filtering, ingress and egress filtering. Ingress filtering is the technique of monitoring incoming packet data. It is considered the first-line of defense in a network because it blocks out unwanted inflow traffic to the network. While this isn't a robust and complete form of defense, it's beneficial because it can greatly reduce the load on some proxy or firewall, and it's an effective for getting rid of the majority of unwanted traffic. Egress filtering is the technique of monitoring the flow of outbound network traffic and prevents any outbound connections to potential threats/unwanted hosts. Egress filtering can be used to disrupt malware, block unwanted services, and gives greater awareness of network traffic. First, we will setup a simple application and document the vulnerabilities. Next, we will implement pod security policies and document the results. Then we will implement network filtering on the base application without pod security policies and document the results. Finally, we will combine both layers of security and document the results. Hopefully, the final application will be free of vulnerabilities.

2. MOTIVATION

It is important to continue researching how network filtering can lead to better application security, especially with a lot of technologies and applications all moving into the cloud. A strong initial filter that effectively identifies a majority of attacks dramatically decreases the chance of a successful attack. If most of the attempted attacks are initially thwarted, there are few attacks that can get through the security in place. A good network filter means less attacks an application has to be wary of, which should lead to a higher defense rate.

3. OUR ARCHITECTURE

Creeate a Kubernetes Cluster with a single pod that has and nginx server running in a container. The container is configured to be a normal user without root priveleges.

Without Pod Security Policies: The user can escalate priveleges to run as root.

With Pod Security Policies: A restrictive PSP should prevent privelege escalation. A violation of the PSP will prevent

the pod from running.

With Envoy: Envoy is to be used alongside the application that we are testing and will be used to monitor the inflow and outflow of traffic. With envoy, we will be able to see how well the implemented pod security policies are working.

4. EXPERIMENTAL RESULTS

First, we will improve the base security of the application through Pod Security Policies. Then we will add another layer of security through Data Plane filters in order to reduce the attack surface and prevent suspicious activity. These extra layers of security should hopefully eliminate the vulnerabilities of the application.

5. RELATED WORK

`https://www.researchgate.net/publication/324562008_Threats_and_Vulnerabilities_of_Cloud_Computing_A_Review`
The article talks about how more and more enterprises are moving their workloads onto the cloud and while security has evolved over time, is still a major concern. The paper goes into details about the various forms of threats and vulnerabilities of the cloud, specifically listing and detailing 17 threats. This paper provides a good foundation for understanding common threats that exploit cloud vulnerabilities.

`https://www.researchgate.net/publication/267691532_MODERN_NETWORK_SECURITY_ISSUES_AND_CHALLENGES`
This paper discusses the threats that networks face and the current network security practices to counteract these attacks. The paper begins by detailing security attacks, security measures, and security tools. The paper goes into great detail about different security methods, such as application gateways and packet filtering. The paper discusses different things that organizations can do to prepare for these attacks and the various technology options.

`https://www.researchgate.net/publication/289756317_Security_Threats_on_Cloud_Computing_Vulnerabilities`
This paper further discusses the vulnerabilities of cloud computing services. The paper details cloud service models and talks about the 3 layers of cloud computing: system layer (IaaS), platform layer (PaaS), and application layer (SaaS). The paper then analyzes the various security issues that each layer faces and talks about the threats that exploit those vulnerabilities.

`https://www.researchgate.net/publication/334548954_Cloud_Security`
This paper discusses the advantages of using cloud services and also reveals the dangers and risks of those services.

`https://nvd.nist.gov/vuln/detail/CVE-2019-5736` This is a known vulnerability, CVE-2019-5736 [1], that allows attackers to execute commands as root within two types of containers, a new container with an attack-controlled image and an already existing container that an attacker has had access to in the past.

`https://cyber-defense.sans.org/resources/papers/gsec/packet-filter-basic-network-security-tool-100197` This article goes into detail about what packet filtering is and how it is used as a network security tool. The paper details the benefits of packet filters and gives a simple implementation of it and discusses the limitations of packet filtering.

6. CONCLUSIONS

7. REFERENCES

[1] Cve-2019-5736.