WIZ

2023

Cloud Vulnerability Report

A comprehensive report on vulnerabilities in cloud environments

By Merav Bar and Amitai Cohen

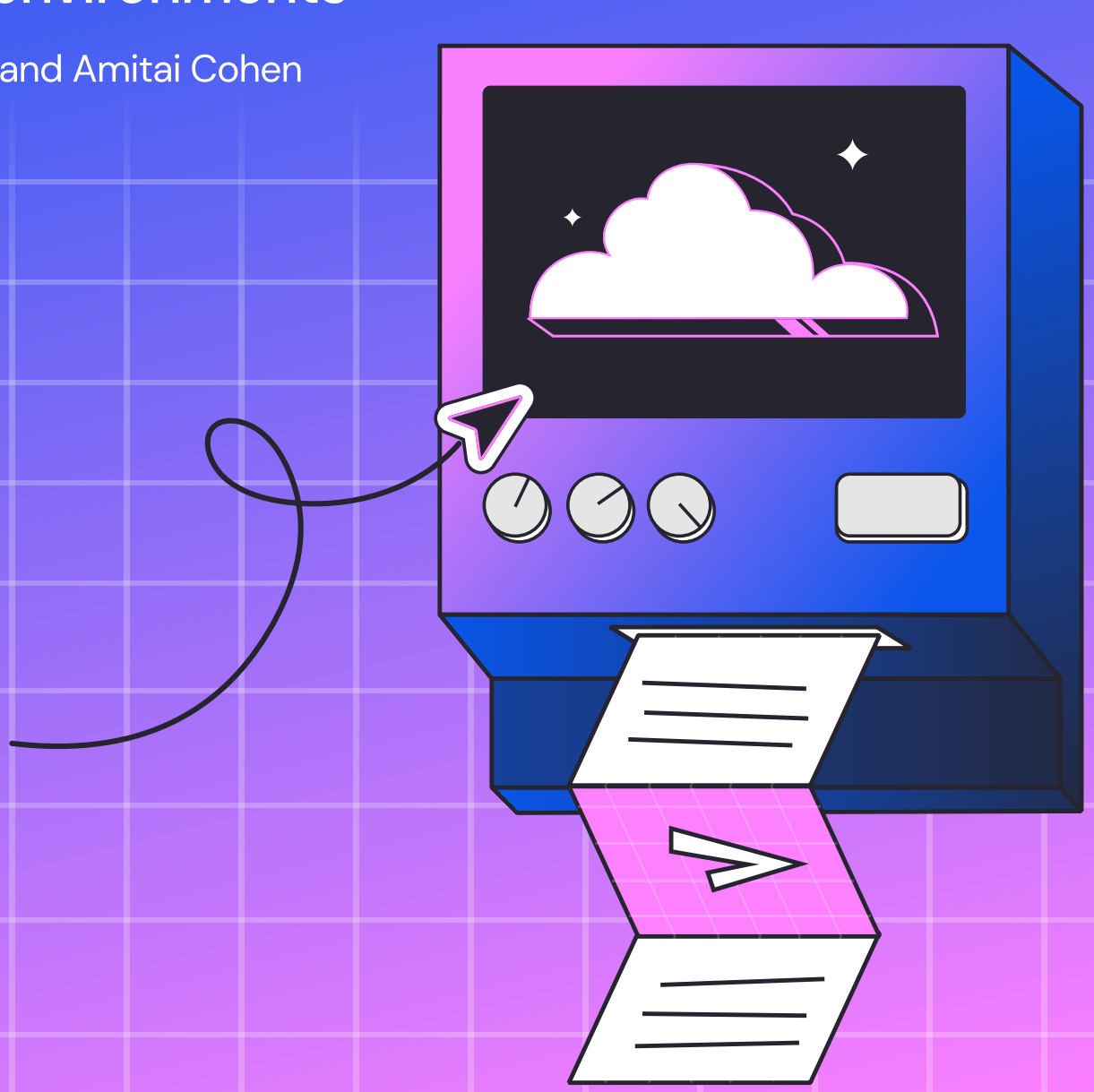


Table of Contents:

Introduction

Vulnerability management in the cloud sits at the critical intersection of AppSec and CloudSec, requiring an understanding of both in order to be effective. The cloud presents us with many new opportunities for vulnerability management, but our approach must take into consideration the unique aspects of cloud environments. In this report we'll present our insights on the subject and discuss the methodology we use at Wiz for incorporating vulnerability intelligence into our triage process, enabling us to help our customers make the best use of their time.

Vulnerability management in cloud environments

Characteristics of vulnerability management in the cloud

The cloud presents several new opportunities for vulnerability management, including:

1. Easier-to-reduce attack surfaces through network controls and image minimization.
2. Serverless and SaaS solutions with faster patch cycles than on-prem.
3. Agentless tooling, allowing for simpler vulnerability detection at scale.
4. Less impactful vulnerabilities in 3rd-party software compared to on-prem (as we shall explain).

Although there are many tools that provide visibility into the vulnerabilities in our environments, there's still a lot of noise to deal with in the form of a seemingly endless flow of CVEs to triage.

Technology prioritization

When analyzing vulnerabilities in the cloud, it is necessary to consider the technologies that run in the cloud and the attack surfaces they expose to potential attackers. In the cloud, we typically expect to see many instances of the following platforms:

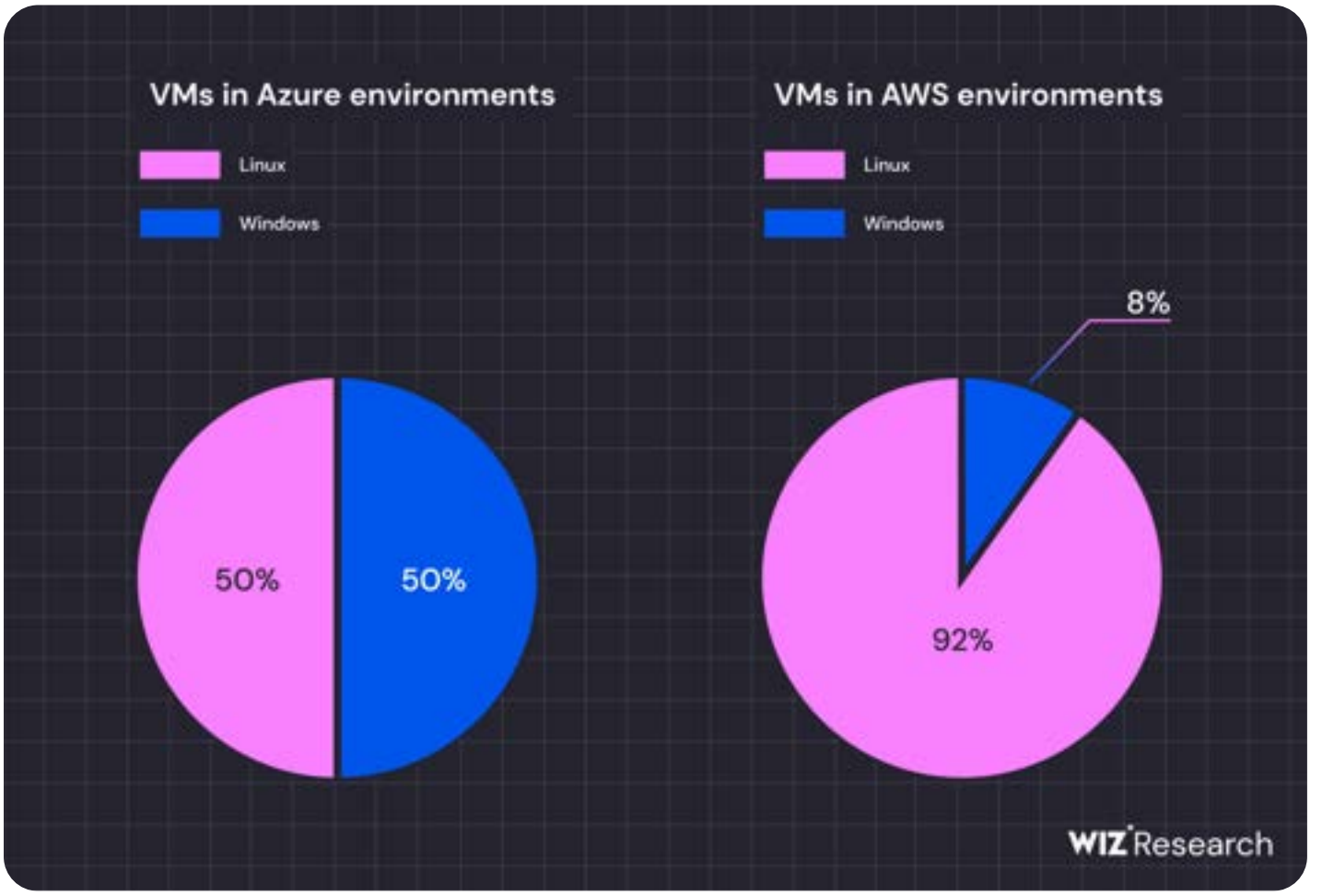
1. Servers (e.g. EC2)
2. CDNs, proxies, and load balancers
3. K8s clusters
4. Containers (on servers/container services)
5. Serverless functions
6. VMs

Conversely, there are many vulnerabilities known to affect components that we simply don't expect to see in the cloud, such as routers, IoT devices, physical network appliances, and phones. Therefore, we can usually safely ignore these vulnerabilities, regardless of their severity.

Similarly, there are many well-known vulnerabilities that might be useful for attacking on-premises machines through phishing, such as vulnerabilities affecting Outlook or Office that require user interaction (e.g., CVE-2023-36893). However, these types of vulnerabilities are much less useful for gaining access to cloud environments, which are mainly made up of server applications that mostly don't involve clicking on things, with the exception of virtual desktops.

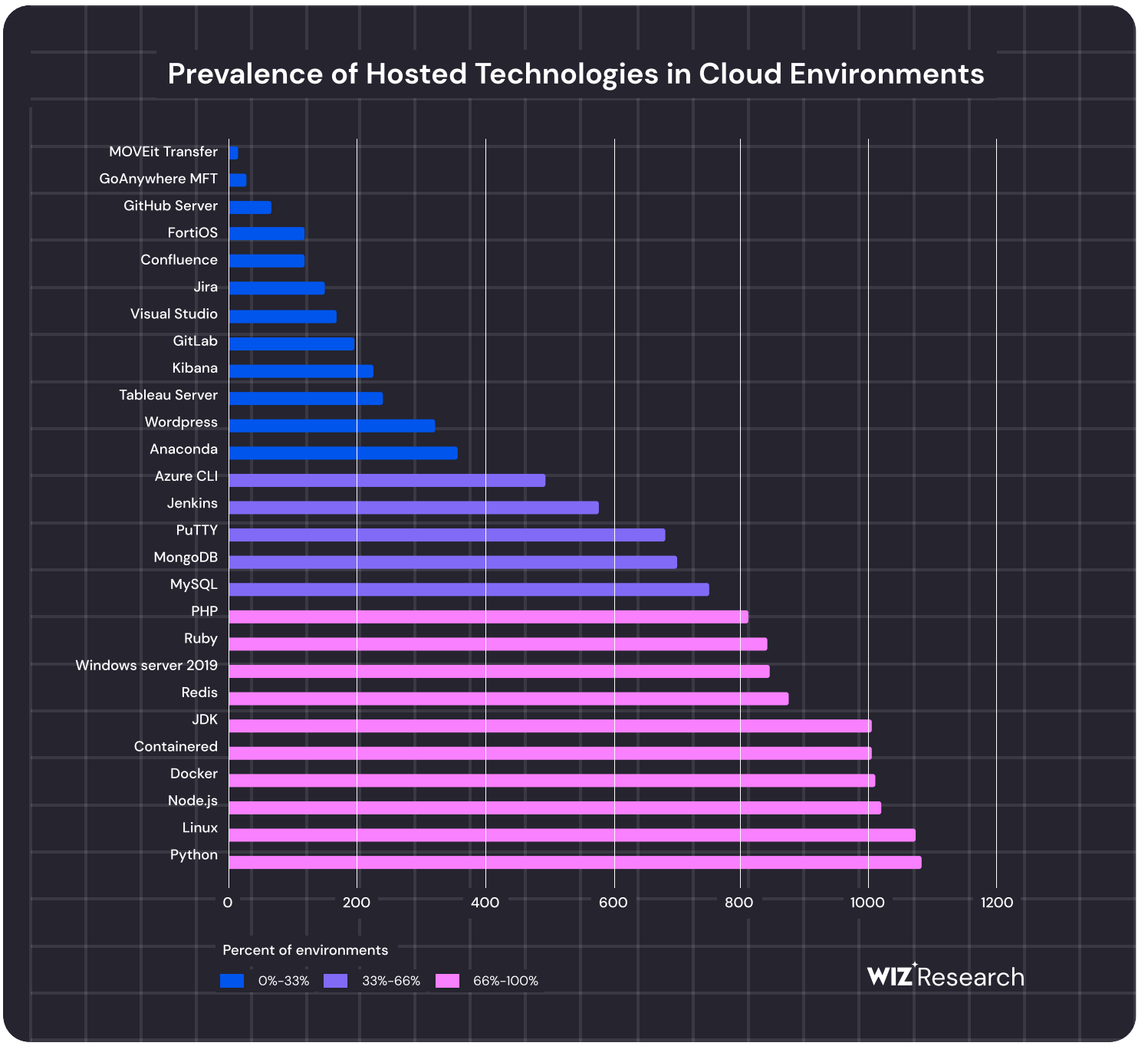
Other vulnerabilities might affect prevalent technologies, such as Vim, that are simply not likely to be exposed to the Internet and therefore have reduced exploitability in cloud environments.

Another factor to consider when prioritizing vulnerabilities is what operating systems run in cloud environments. Our data shows that the most prevalent is Linux, which is to be expected. However, there are differences between each of the major cloud providers. For example, our data shows that the total number of Windows VMs in Azure and AWS are about the same, but in Azure there's an even split between Windows and Linux, whereas in AWS Linux outnumbers Windows 10:1.



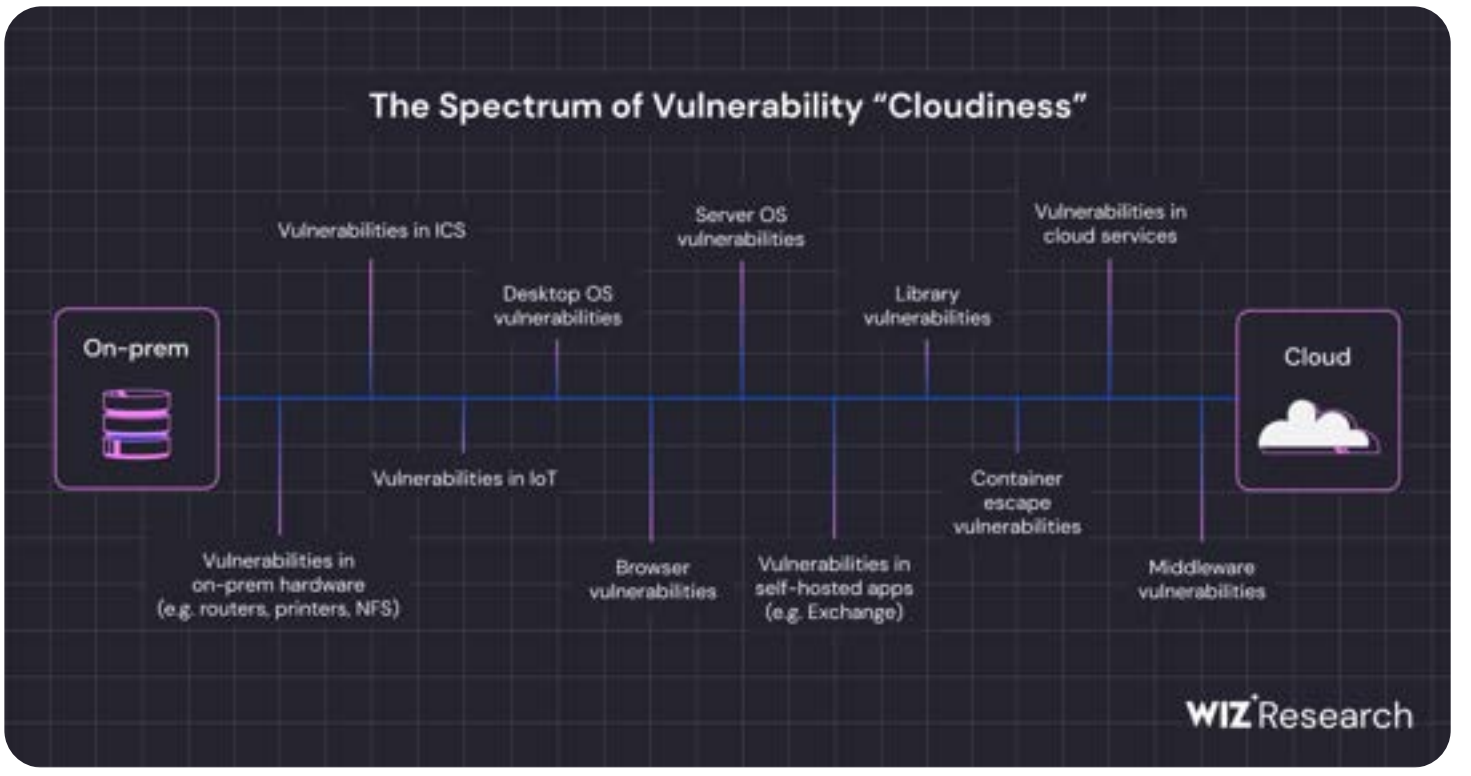
Regardless of operating system, when it comes to software running in the cloud, many popular technologies can run on "anything," such as Python or Node.js, which naturally makes them highly prevalent, leading to the vulnerabilities affecting them being prevalent in turn. The cloud is also home to a great many publicly-exposed web apps, management portals (like Jenkins), database management systems, admin consoles, file scanners, and web crawlers — each of which creates significant attack surface that requires management.

However, many cloud customers prefer the use of managed software, especially when it comes to products like Confluence or Github. These are (perhaps surprisingly) less prevalent as hosted technologies thanks to the popularity of their managed SaaS counterparts, and their vendors update them as soon as a patch is available, thereby relieving customers of vulnerability management.



When discussing the prevalence of different technologies, in our own research it is important to consider the percentage of affected environments rather than the total number of workloads. This approach accounts for variations in the size of cloud environments, which can range from massive to minuscule depending on the organization. Additionally, certain technologies may simply be rarer by design. For instance, most organizations don't have more than one Confluence server, whereas the same Python library might be observed on the majority of workloads in any given environment.

By focusing on the 5,000 most prevalent vulnerabilities across all cloud environments we've analyzed, we can gain a better understanding of this landscape. Accordingly, we've placed vulnerabilities affecting different types of technologies along a spectrum, ranging from what we consider to be the least to most relevant to security teams in charge of cloud environments:



This spectrum can help security teams assess the relevance of any given vulnerability. For example, CVE-2022-29149 is a vulnerability affecting OMI, which is cloud middleware utilized in Azure environments. That would make it very relevant in the cloud, and patching it should be a high priority for cloud security teams. In fact, our data shows that almost 40% of cloud environments have at least one publicly exposed workload affected by a vulnerability in middleware or other CSP-provided software.

Conversely, CVE-2020-15683 is a vulnerability in the Firefox web browser, which might be very prevalent in cloud environments (since it's preinstalled in some Linux distributions), but exploiting the vulnerability requires that a user visits a malicious website. Most servers would therefore not be susceptible, making this vulnerability a lower concern in cloud environments.

Vulnerability type prioritization

Beyond narrowing our long list of CVEs based on the technology they affect, we also need to consider the types of vulnerabilities that are most useful to attackers targeting cloud environments.

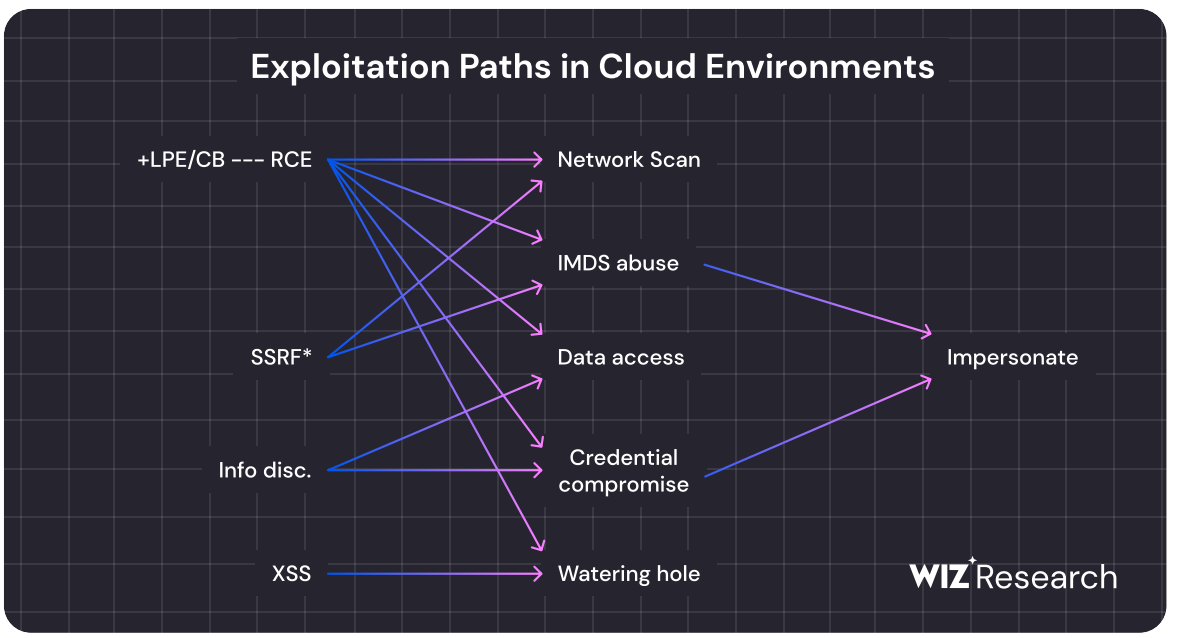
To this end, we can list the following likely goals of such threat actors:

1. Steal sensitive data from cloud workloads
2. Conduct a supply chain attack against an organization's customers
3. Hijack an organizations' resources (e.g., for cryptomining)
4. Compromise credentials from an organization's workloads to facilitate lateral movement
5. Impersonate cloud workloads to achieve any of the other goals above

Depending on what an attacker is trying to achieve, some types of vulnerabilities will not be useful at all, others will be sufficient, while others might prove to be overkill.

All an attacker usually needs from a publicly exposed machine are cloud keys for lateral movement and identity-based privilege escalation, which they can often obtain either by stealing secrets from environmental variables or connecting to the IMDS. This means that in many cases they only really need an information disclosure vulnerability. Therefore, local privilege escalation (LPE) vulnerabilities — and sometimes even remote code execution (RCE) vulnerabilities — could very well be redundant in this scenario.

In other words, while attackers targeting on-prem environments often aim for persistence and local privilege escalation, those targeting cloud environments can achieve their goals with less powerful vulnerabilities — in some cloud environments, an attacker could reach total account takeover with just an SSRF. The significance of SSRF in cloud environments can also be apparent in special bug bounty programs devoted to discovering them, such as Azure's SSRF Research Challenge.



However, sometimes a threat actor will need something more. For instance, in public multi-tenant environments, an attacker can easily create an account, authenticate and sometimes even execute code by design, making privilege escalation the next logical step. Moreover, there is one class of LPE vulnerability that can be very useful to an attacker aiming to move laterally and cross tenant boundaries: container escape. This risk is quite prevalent, with our data indicating that 20% of cloud environments have at least one publicly exposed container host affected by a container escape vulnerability.

Key questions for vulnerability triage

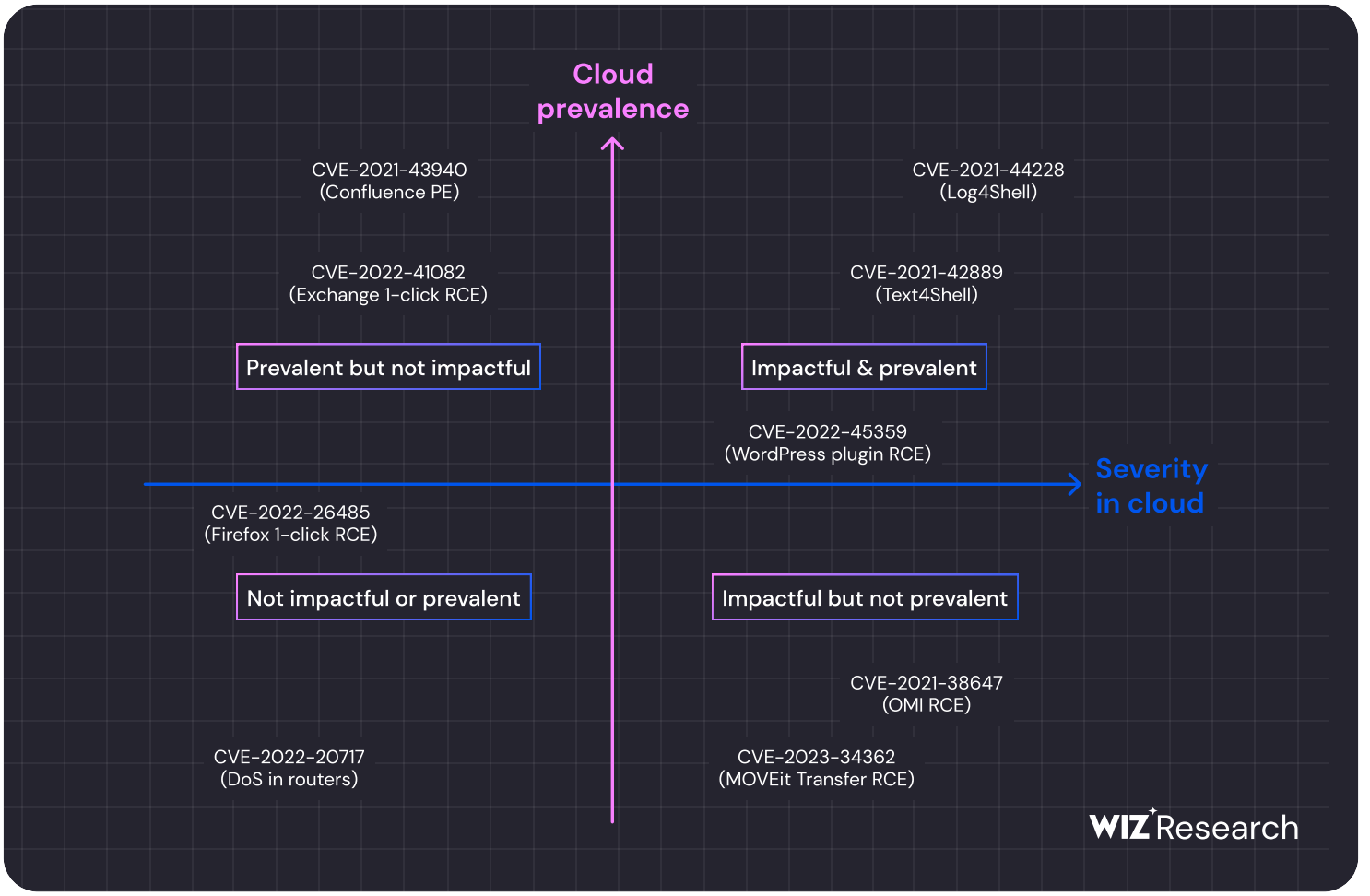
Based on all of the above, we can develop an effective methodology for vulnerability triage in cloud environments, made up of the following key questions:

1. What is the "cloud value" of the affected technology to an attacker?
    a. How prevalent is the affected technology in cloud environments?
    b. Is it likely to contain sensitive data?
    c. Is it usually granted high privileges in the environment?
2. What is the initial access potential granted to an attacker?
    a. Does the vulnerability allow arbitrary code execution on the workload?
    b. Does it allow access to data on the workload?
3. Are there significant prerequisites for exploitation? (e.g., prior access, post-authentication, non-default configuriation, user interaction, etc.)

Putting theory into practice

The previous key questions enable us to build a model for estimating vulnerability impact in the cloud, as demonstrated in the graphic below. This analysis helps us differentiate between vulnerabilities that are justifiably promoted and those that are overhyped, at least in the context of cloud environments.

Vulnerabilities in technologies that are rare in the cloud (and also don't really help attackers gain initial access) are shown on the bottom left quadrant, such as router Denial-of-Service (DoS) vulnerabilities. On the other hand, highly useful vulnerabilities in popular, privileged, or data-rich technologies are shown in the top right (e.g. Log4Shell). Security teams should focus their attention on these vulnerabilities in particular.



As we begin to work through triaging the many CVEs in our cloud environment, we can apply many filters to further reduce the number of CVEs worth prioritizing. No single filter works well on its own, but a strong combination of filters can be highly effective.

Step 1: Leveraging CVSS metrics

First, we utilize CVSS metrics, which provide a severity score and basic information about exploitability conditions and impact type for each vulnerability. They enable us to focus on vulnerabilities with the highest potential for initial access to cloud environments. Specifically, we prioritize critical or high-severity vulnerabilities with characteristics like network attack vectors, no requirements for user interaction or prior privileges, and high integrity or confidentiality impacts (i.e.an attacker has read/write capabilities). The combination of these criteria usually corresponds to either remote code execution (RCE) or information disclosure (in contrast, availability impact usually corresponds to Denial-of-Service, which is of less concern in most scenarios).

Step 2: Integrating vulnerability intel sources

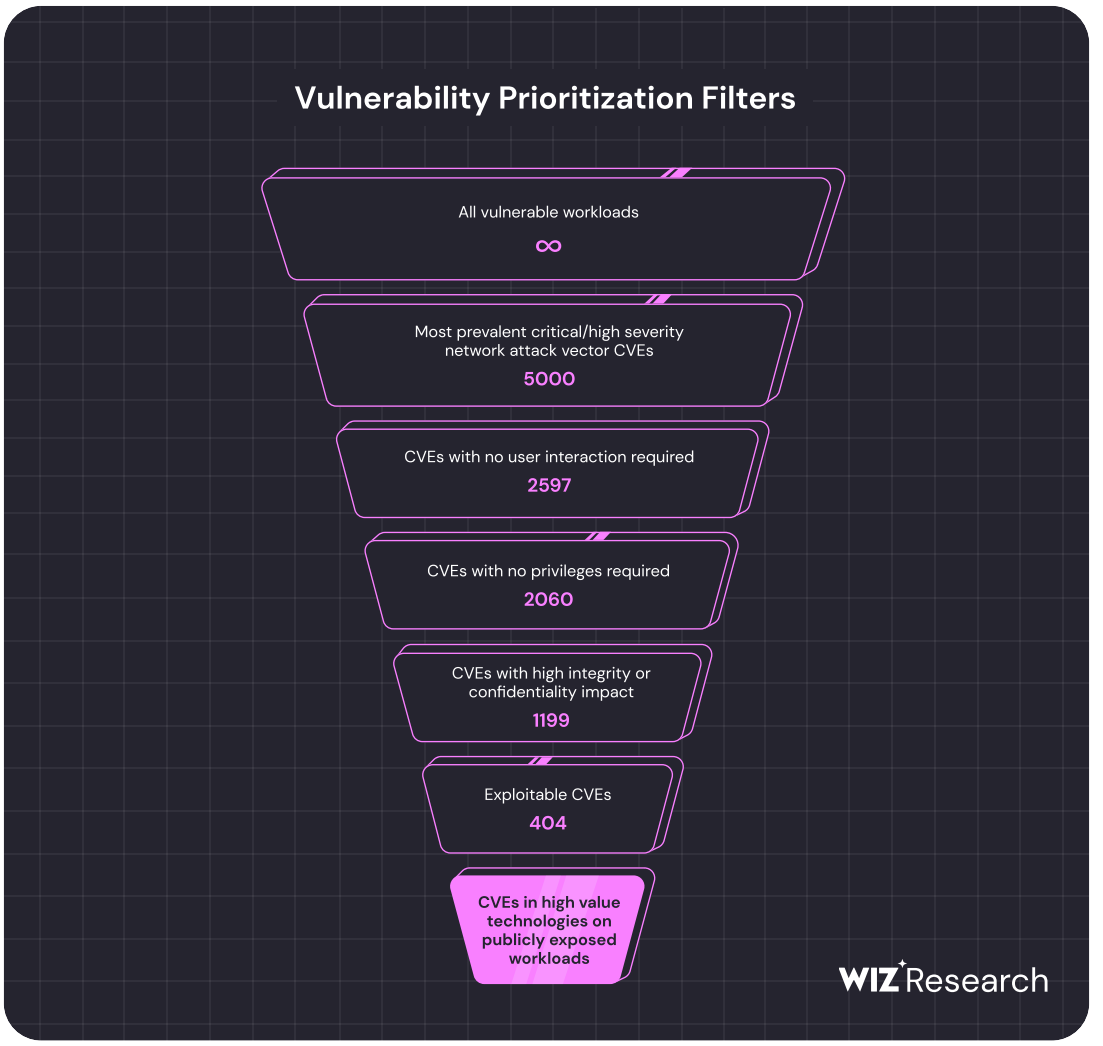
In addition to CVSS metrics, we incorporate external vulnerability intelligence sources such as various exploit databases and CISA KEV. Furthermore, we leverage EPSS (Exploit Prediction Scoring System) to obtain theoretical exploitability likelihood scores. By analyzing this information, we can focus exclusively on vulnerabilities that are most likely to be exploited in the wild.

Step 3: Utilizing technology prevalence data

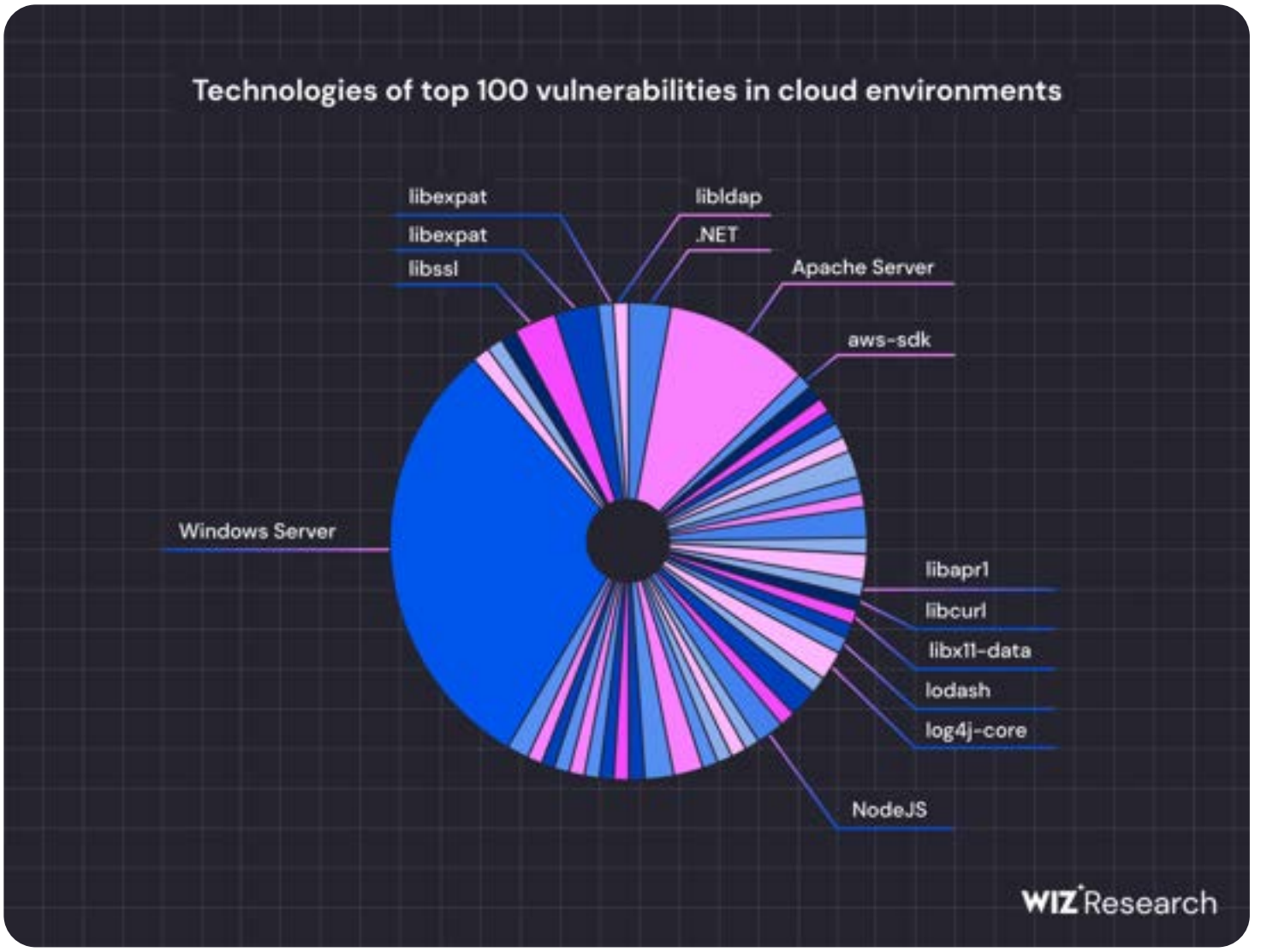
Our own data on technology prevalence in cloud environments becomes instrumental in prioritizing vulnerabilities in high-value technologies. By considering the widespread usage of certain technologies, we can allocate resources to address vulnerabilities that pose the greatest risk in cloud environments, while deprioritizing vulnerabilities in technologies that aren't prevalent in the cloud.

Final Results

After applying these filters, we are left with approximately 400 vulnerabilities out of our original 5,000, representing approximately 8% of the initial group. These remaining vulnerabilities can be considered the creme-de-la-crème of critical/high-severity network vulnerabilities.



The next step involves prioritizing these vulnerabilities based on their “cloud tech value” by focusing on vulnerabilities in high privilege, data-rich software. To this end, we can investigate what technologies are affected by the top 100 most prevalent vulnerabilities.



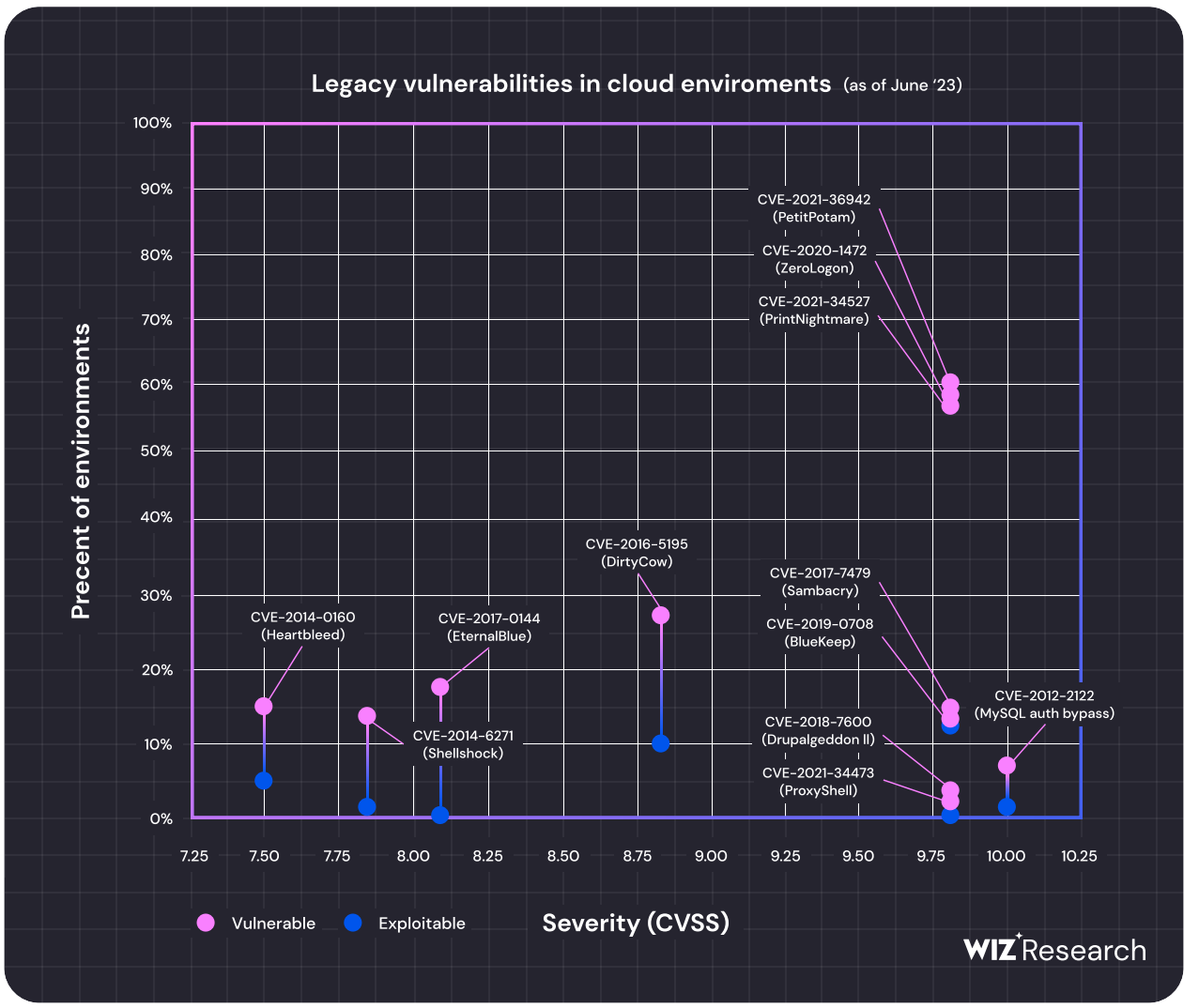
This breakdown helps us gain valuable insight into the vulnerabilities and overall attack surface of the cloud. First, it highlights the vulnerabilities that cloud customers choose to ignore — otherwise they wouldn’t be as prevalent as they are. Each of these vulnerabilities seems to be considered overrated by many cloud customers, or perhaps these customers have simply decided to apply mitigations and workarounds rather than patching these issues.

Several of the most prevalent vulnerabilities in cloud environments are found in just two libraries: OpenSSL and Expat (XML parser). The impact of vulnerabilities in these libraries largely depends on the circumstances of the affected workload and how the library is being used in practice. Many application developers opt to use their own versions of these libraries, which may or may not be affected by the same vulnerabilities. This means that we might find several different versions on the same workload with only one of them in use.

The graph above also reveals that Windows Server has a significant number of prevalent CVEs. We believe this isn't necessarily indicative of the level of security of Windows Server, but rather should be attributed to Microsoft's relatively high level of transparency and the widespread usage of the Windows operating system. Both attackers and security researchers focus their efforts on finding vulnerabilities in Windows, resulting in a higher number of reported vulnerabilities overall. Finally, we can easily determine that Log4Shell continues to be prevalent, supporting the Cyber Safety Review Board's assessment that it is an "endemic" vulnerability, expected to persist.

Applying public exposure checks

Beyond triaging the vulnerabilities themselves, security teams should check the public network exposure of vulnerable workloads. This allows us to assess their real-world impact on a given cloud environment depending on their remote exploitability. By validating public exposure and conducting precise exploitability checks, the number of effectively exploitable instances tends to drop dramatically. To demonstrate this, it helps to look beyond the recent past and check the prevalence of "legacy" vulnerabilities like SambaCry, EternalBlue, or ZeroLogon. Our data shows that these vulnerabilities have quite a long tail in cloud environments, with nearly 20% of environments still vulnerable to EternalBlue, for example.



However, in most cases these vulnerabilities aren't remotely exploitable. When we check for more precise exploitability conditions and validate public exposure of vulnerable workloads, the numbers drop dramatically. For example, by checking to determine whether workloads affected by SambaCry are publicly exposed on an SMB port, the number of relevant workloads drops to zero. Similarly, by determining whether Windows workloads affected by ZeroLogon are being used as DCs (which is a requirement for exploitation of this vulnerability), the number is reduced from around 60% to 2%. In total, the effective attack surface exposed by these vulnerabilities is quite lower than what was initially apparent.

Summary

Cloud vulnerability management presents us with new challenges and opportunities.

1. The cloud has different "physics," which change the impact of certain vulnerabilities, for better and for worse.
2. We suggest focusing patching and attack surface reduction efforts on vulnerabilities with high initial access potential affecting high-value technologies.
3. No single prioritization filter is good enough on its own, but a strong combination works well.

About Wiz

Wiz secures everything organizations build and run in the cloud. Founded in 2020, Wiz is the fastest-growing software company in the world, scaling from $1M to $100M ARR in 18 months. Wiz enables hundreds of organizations worldwide, including 35 percent of the Fortune 100, to rapidly identify and remove critical risks in cloud environments. Its customers include Salesforce, Slack, Mars, BMW, Avery Dennison, Priceline, Cushman & Wakefield, DocuSign, Plaid, and Agoda, among others. Wiz is backed by Sequoia, Index Ventures, Insight Partners, Salesforce, Blackstone, Advent, Greenoaks, Lightspeed and Aglaé. Visit https://www.wiz.io/ for more information.