HADRIAN

# Mapping Cyber Risks from the Outside

2024 Report

# Table of Contents

# Introduction

Defense-oriented cybersecurity strategies have historically been favored to protect organizations' digital assets. This approach has its roots in the castle-and-moat strategy, which was utilized in the 1990s and early 2000s to protect a small number of known assets. However, defensive strategies are inherently reactive and are increasingly putting organizations' security postures at risk.

Today security teams encounter numerous business challenges. IT and development teams have grown and are many times larger than security departments resulting in a pace of change that is hard for security teams to keep up with. Technology environments have also changed dramatically, with 3rd party software increasingly utilized and even core infrastructure moving to cloud, increasing the scale and complexity of what security teams must secure.

Threat actors have taken notice and there has been a steady increase in the number of attacks targeting digital assets that are exposed to the internet. This trend requires security teams to view their attack surface from the hacker perspective.

This report contains the key security risks that Hadrian has observed over the last 12 months, highlighting why continuous and comprehensive monitoring of the attack surface is necessary. Our vision is to empower organizations to operate securely by advocating for a platform-driven, holistic approach to offensive security.

# Key Findings

## 1. Continuously Assess the Attack Surface

Conduct regular, continuous assessments of the organization's attack surface to identify and address new exposures promptly.

New risks emerge on a daily basis, and without ongoing assessments, vulnerabilities may go unnoticed, increasing the likelihood of compromise.

## 2. Monitor DNS Infrastructure for Misconfigurations

Establish continuous monitoring of DNS infrastructure to detect and swiftly remediate misconfigurations.

CNAME record misconfigurations account for over 25% of risks discovered by Hadrian and can be exploited in many ways, such as phishing or chaining with other issues, making real-time monitoring essential.

## 3. Continuously Test Web Applications for Injections

Regularly assess web applications for injection vulnerabilities to address risks as they emerge.

Injection risks represented nearly 60% of all critical severity risks found by Hadrian over the past year, highlighting the importance of proactive testing.

## 4. Incorporate Context into Severity Scoring

Implement vulnerability scoring methodologies tailored to asset context and organizational risk priorities.

Many of Hadrian's highest severity risks lack associated CVEs, and CVSS base scores alone do not accurately reflect the potential impact, making context-driven prioritization critical.

## 5. Prioritize Limiting Information Leaks

Focus on remediating exposed secrets and injection risks to reduce information leaks.

Over 67% of application and service exposure risks have low or informational severity, but threat actors frequently exploit leaked credentials and RDP servers, requiring action on sensitive information leakage.

## 6. Improve Collaboration for Faster Remediation

Establish clear communication channels and urgency indicators for developer and tech teams to address misconfigurations and injection risks.

These risks take an average of 70 days to resolve, which takes 3 times longer than simpler issues, and their timely remediation requires better cross-functional collaboration.

# Verified Risk Categories

26.9%

### Domain Name Servers

Including domain name takeovers, CNAME issues, and DNS record issues. DNS vulnerabilities are typically exploited for domain takeovers.

22.1%

### Service Exposure

Remote Desktop Protocol (RDP), Virtual Network Computing (VNC), SSH, or FTP and are usually on an open port.

15.4%

### Injection

Malicious code like SQL, template injection, Cross-Site Scripting (XSS), Local File Inclusion (LFI), or Remote Code Execution (RCE).

12.2%

### File Directory Exposure

Allow access to files or directory structures that let an outsider gain insight into the technical architecture or key information on the system.

9.5%

### Misconfiguration

Lead to data being exposed about the application or the data that the application has access to. Examples include request smuggling and GraphQL introspection.

7.9%

### Application Exposure

When applications leak information due to internal logging being exposed on the internet, such as status pages, leaking logs, or memory dumps.

1.9%

### Authorization & Authentication

Problems including bypassing authentication, Insecure Direct Object References (IDOR), Cross-Site Request Forgery (CSRF).

1.2%

### Cloud & SaaS Configuration

Issues related to the configuration of Cloud and Software as a Service (SaaS) platforms.

0.6%

### Exposed Secret

Any kind of exposed key, token, secret, or sensitive credentials.

3.3%

### Other risks

Any kind of exposed key, token, secret, or sensitive credentials.

# Risks Discovered in the Attack Surface over Time

The rapid pace of technological change that many organizations experience can introduce new risks into the attack surface. This is demonstrated by the graph to the left which shows consistent discovery of new risks on a monthly basis.

After an initial onboarding period, the risk discovery for the majority of organizations reaches a steady state. This indicates a need for continuous monitoring of the attack surface for new vulnerabilities.

# Severity Scores of Attack Surface Risks

Hadrian prioritizes validated risks using a context-based stakeholder-specific vulnerability categorization methodology. Risk scores are based on the automated determination of business relevance and attractiveness of assets to attackers, along with discoverability, impact, and ease of exploitation. The composite score based on these factors is then calculated and presented to customers.

For London Business School we found a risk that would be classified as Medium in CVSS 3 and was scored as High risk by Hadrian because of the asset context. The platform identified that the asset where the risk was found on was load-balanced, and load-balanced means it is likely high traffic, and high traffic means it is likely to be important to the school's operations.

### Critical

Indicates that a risk needs immediate attention. For example, an SQL injection vulnerability that leads to access to data on an important domain.

### High

Indicates that a risk should be urgently fixed. For example, source code disclosure that can lead to insights into the inner of an application.

### Medium

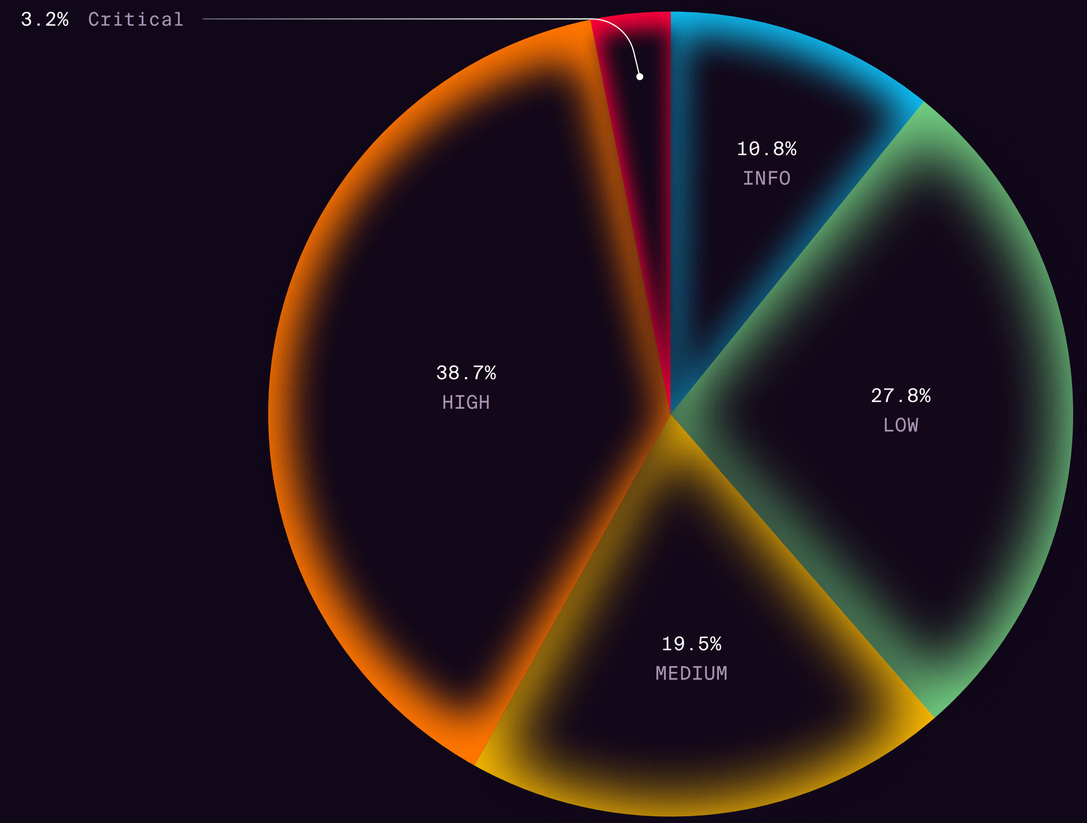
Indicates that a risk should be planned to be fixed. For example, a redirect vulnerability that can be used in phishing.

### Low

Indicates that a risk does not have a high impact but could be fixed for security hygiene purposes. For example, exposed application metrics.

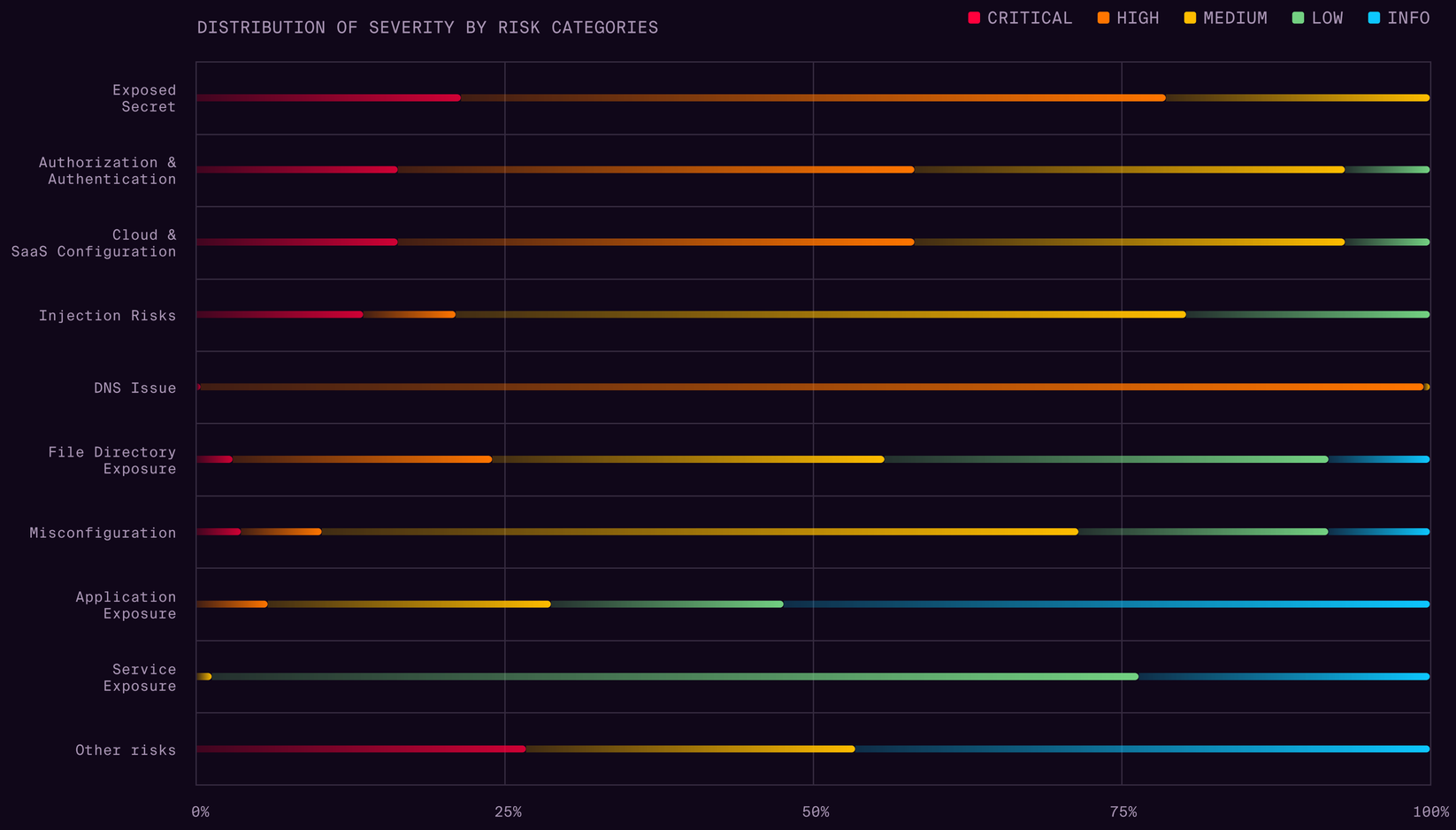
### Info

Indicates that an issue can be fixed following best practices. For example, a WordPress Readme file was exposed in one of your systems.



# Distribution of Risks

Focusing on verified risks and using a context-based severity scoring methodology reveals that the exposed applications and services are not the biggest priority for security teams. Hadrian's research has found that these easily detectable risks are less likely to result in an incident which could mislead security teams into prioritizing the wrong risks. In contrast, Injection Risks, Authorization & Authentication issues, Exposed Secrets, and Cloud & SaaS Configurations typically have higher severity scores indicating a greater risk to an organization that should be remediated first.



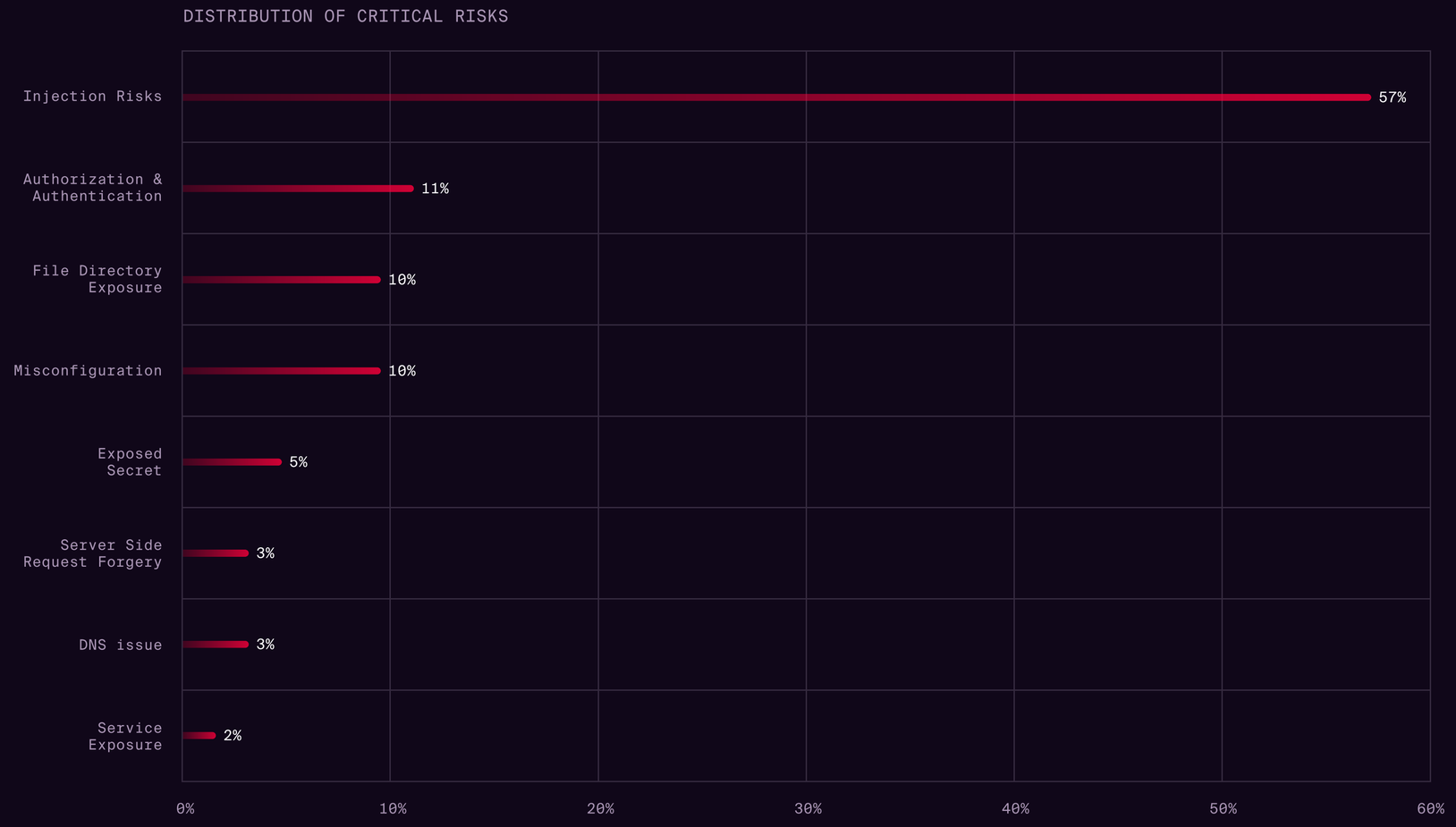
# Analysis of Critical Severity Risks

Injection vulnerabilities, including SQL injection, Cross-Site Scripting (XSS), and other code injection risks, account for nearly 60% of critical severity risks that are exploitable in the attack surface. These risks have been rated critical as they allow attackers to manipulate data, access restricted areas, or run malicious scripts within applications.

As businesses expand, they naturally deploy more web applications, APIs, and cloud services to support operations. Without proper input validation and security practices, these can become easy targets for attackers.

Despite awareness of threats, developers may deploy new applications using legacy code or insufficient security controls, exposing systems to Remote Code Execution (RCE) or Local File Inclusion (LFI) attacks. APIs, crucial for integrating internal systems and external services, add further risk. If not properly configured, APIs can become conduits for command injection attacks, providing attackers with access to sensitive data or backend systems.

As injection risks can be introduced at any time. It is therefore recommended that organizations continuously monitor their production systems.



# Breakdown of Injection Risks

### Cross-Site Scripting (XSS)

XSS is the most common injection risk accounting for 38.3%. It  occurs when malicious scripts are injected into a trusted website and executed in the browser of other users. This happens when web applications fail to properly sanitize user input or display it directly on the page, and allows attackers to steal user data, manipulate website content, or perform actions on behalf of users without their knowledge. XSS is ranked second in MITRE's Top 25 Most Dangerous Software Weaknesses which makes the high occurrence rate a concern.

### Open Redirect

This is a technique where users are redirected to unintended websites, often for phishing or malware distribution. It happens when web applications accept a user-provided URL for redirects and fail to validate or restrict it, and leads to phishing attacks, malware downloads, or loss of user trust in legitimate websites.

Despite it being the second largest category of injection, attack detected it is not found in MITRE's Top 25 Most Dangerous Software Weaknesses. Open Redirect vulnerabilities are dangerous because they can be chained with Oauth bypasses, XSS and other issues.

### Remote Code Execution (RCE)

An RCE injection exploits vulnerabilities in software (like input sanitization flaws) to inject and execute arbitrary code on a target system. It enables attackers to steal data, install malware, or completely take over the system.

### SQL Injection (SQLi)

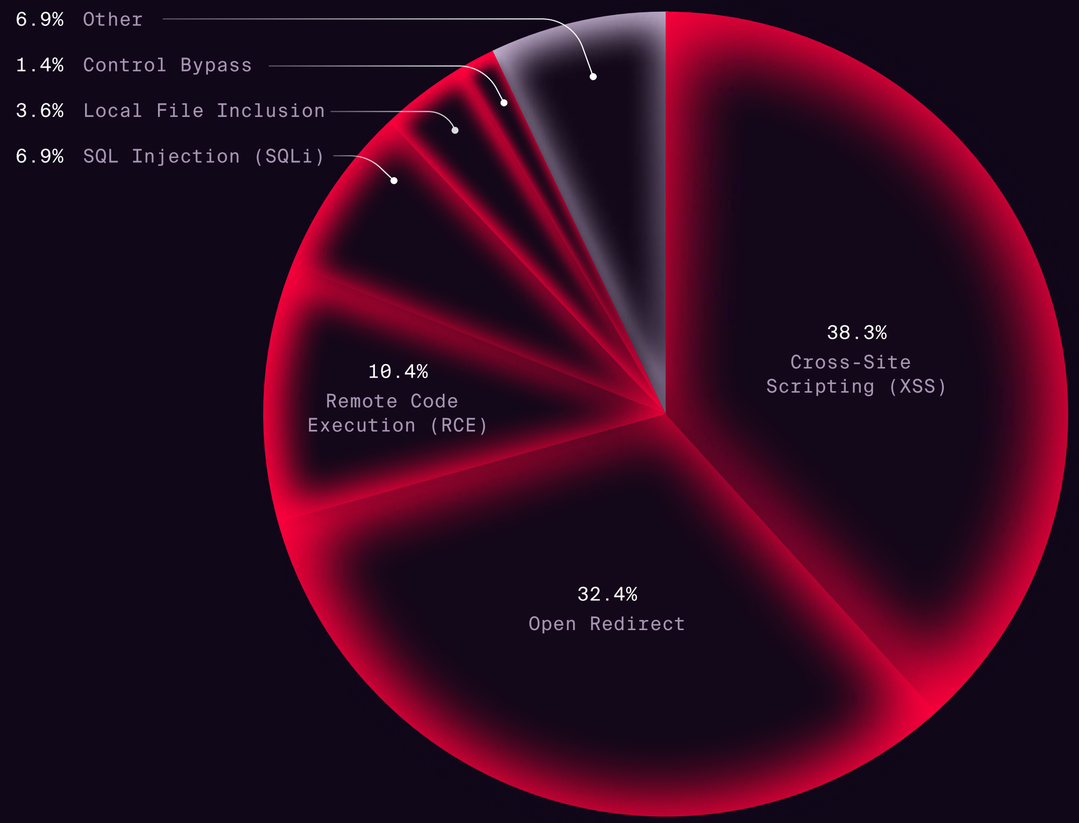
The attack allows malicious SQL code to be executed on a database. When input fields (e.g., login forms) accept user input without proper sanitization, embedding it in SQL queries. Attackers get to access, modify, or delete sensitive data, compromising the entire database.

### Local File Inclusion

This attack exploits a vulnerability to include unauthorized files on a server via web applications and happens when user-controlled input is used to specify a file path while input isn't properly validated. This exposes sensitive files or escalates to remote code execution.

### Control Bypass

This allows attackers to bypass security mechanisms like authentication and authorization, by exploiting flaws in input validation or poorly implemented security checks (e.g., manipulating URLs to skip login pages). This attack grants unauthorized access to sensitive areas of applications, compromising security and data privacy.



# Analysis of High Severity Risks

DNS issues represent a large proportion of the high risks discovered by Hadrian. As an essential component of the internet, DNS servers can be an attractive target for threat actors. The most common DNS issue is dangling CNAME records which leaves domains vulnerable to hijacking. If the linked destination becomes unclaimed or expired, attackers can register it, redirect traffic to malicious sites, or impersonate legitimate services, facilitating phishing and malware attacks.

A single DNS misconfiguration such as dangling CNAME records can result in domain takeovers, where attackers hijack web traffic and redirect it to malicious sites, launch phishing attacks, or even steal cookies scoped to parent domain. Enterprise businesses often utilize multiple DNS servers and configurations, increasing the chance of misconfigurations.

Phishing attacks are one of the most common attack vectors and organizations should take steps to prevent their DNS infrastructure from being utilized by threat actors.

# The Hype Surrounding CVE Risks

In 2020 the number of new CVEs documented was just over 18 thousand, last year it was almost 29 thousand, and it appears that this trend will continue for the foreseeable future. Simultaneously, there has been an increase in the number of exploits that have received significant media attention.

This September, CUPS vulnerabilities were released by security researcher Simone Margaritelli following much discussion only for it to be revealed that the requirements for real-world exploitation were so high that exploitation was unlikely in the vast majority of deployments. Similarly, last year, a vulnerability in the libcurl library was overhyped and the impact was far lower than expected.

Patch management is an important part of managing exposures in the attack surface but they are only a part of it, and many risks such as SQL injections or Cross Site Scripting for the web applications developed by organizations will have no associated CVE. Hadrian's research reveals that the majority of risks that organizations should prioritize are related to web application issues and cloud misconfigurations.

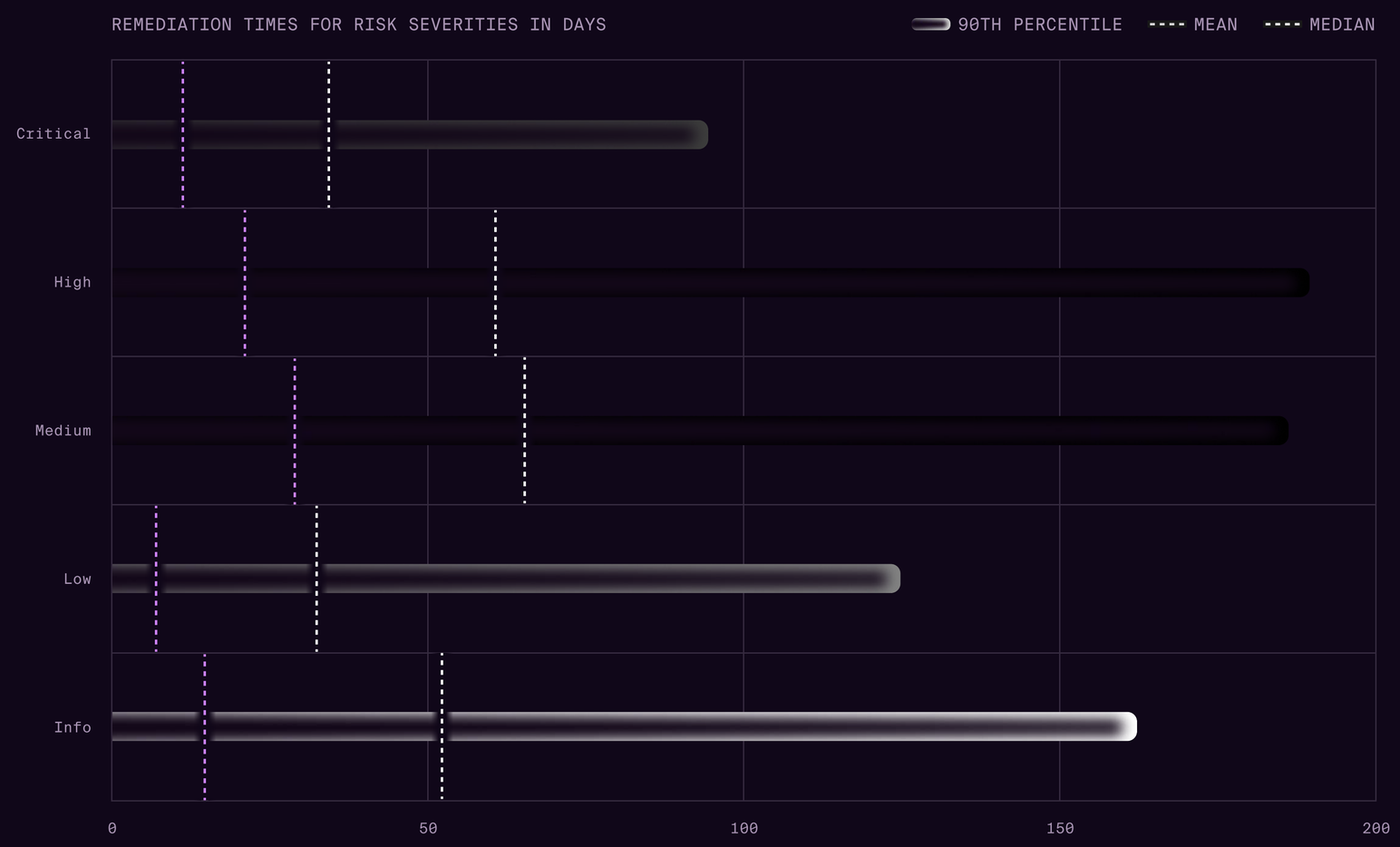
It is important to take a holistic approach to threat exposure management by assessing the full range of possible risks across the entire attack surface. Furthermore, in order to respond quickly to new CVEs organizations should maintain a detailed inventory of their attack surface in order to quickly identify vulnerable systems.

# Remediation Response Times

Timely remediation of discovered risks is essential for minimizing the window of vulnerability for organizations. One of the key metrics used by organizations is the mean-time-to-remediation, which measures the time from discovery to remediation for each risk.

There are several notable features when looking at the remediation times broken down by severity:

- The median time to remediation is approximately one-third of the mean time. This indicates that a minority of risks require exceptionally long times to be remediated, skewing the data. This is backed up by the 90th percentile remediation figure which reveals that it takes half a year to resolve some risks.
- Critical severity risks take longer to remediate than High severity. Hadrian uses a contextual approach to severity scoring which considers the importance of an asset to business operations. Remediating a critical risk can require activity during specific change windows, leading to a longer remediation time in comparison to high-severity risks.
- The remediation time for Low and Info severity risks is shorter than Medium severity. This observation can be explained by the level of effort required for remediation which is often significantly less than Medium risks. For example, it could be as simple as disabling GraphQL introspection on an asset.

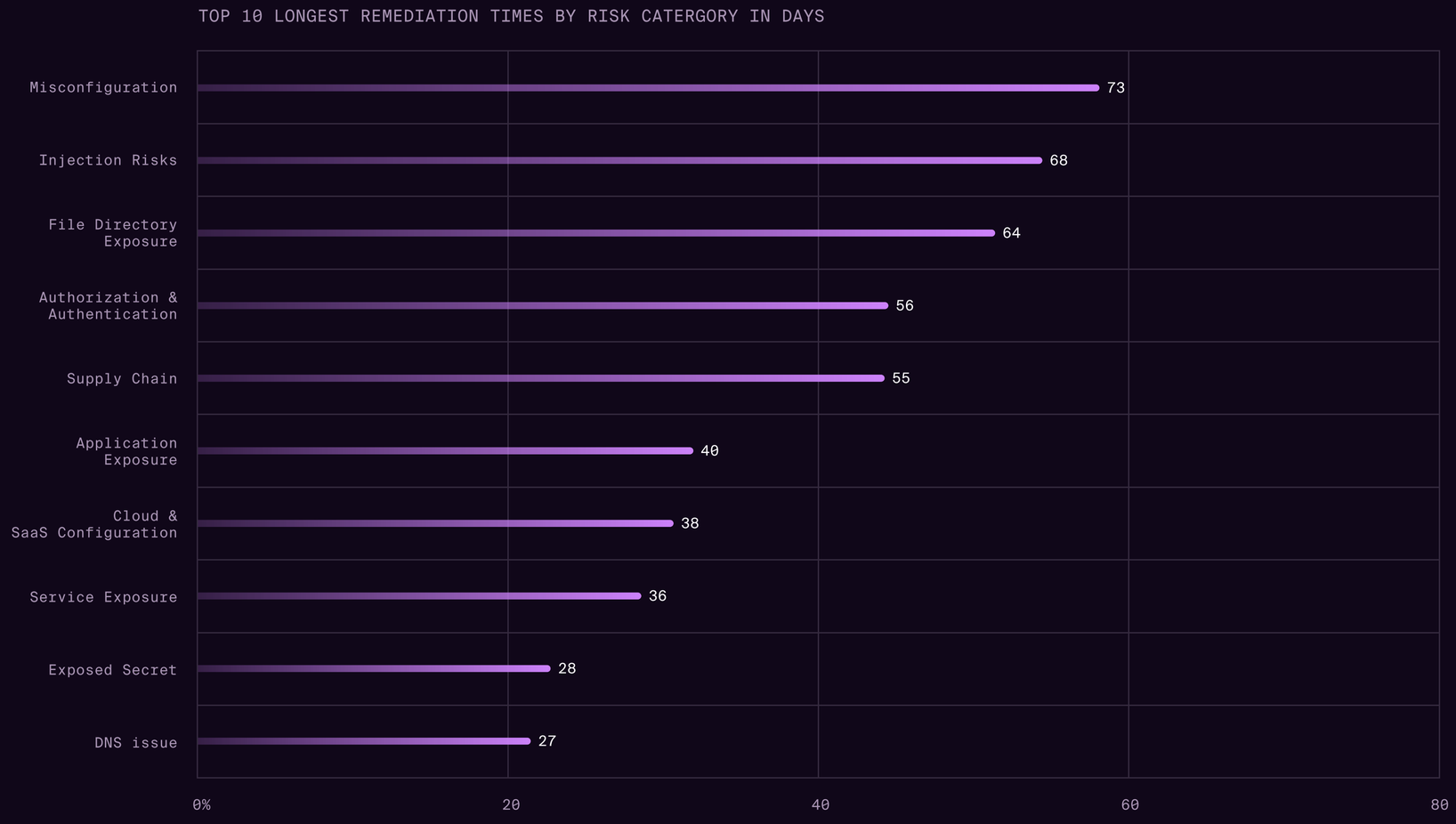


# Remediation by Risk Category

The time required during remediation can vary dramatically depending on the category of risk. Those that can be remediated through more straightforward actions, such as a configuration change, are typically resolved in a number of weeks. Whereas more complex remediation activities, which require developers to fix flaws in web applications, take three to four times longer to resolve.

The category of risk appears to have a significant bearing on time to remediation:

- On average Misconfigurations take the longest category of risk to remediate, typically taking 72.6 days. The remediation of these risks often deals with having to stop the service/web application and redeploy with e.g. an updated version of a library/service, which can delay remediation.
- Injection risks take the second longest time, with a mean of 68 days to remediate. These risks will require web application developers to find and fix issues in deployed software, leading to longer remediation times.
- Application and Service Exposure, and Cloud & SaaS Configuration risks all relate to the unintended exposure of sensitive assets to the internet. These are typically faster to remediate as it can be done by changing system settings in order to restrict access.



# Recommendations

### Monitor DNS infrastructure for misconfigurations

DNS Issues, in particular CNAME record misconfigurations, are the most common risk discovered by Hadrian, accounting for over a quarter of risks. These could be leveraged in phishing attacks to compromise an organization. Continuous monitoring of DNS infrastructure should be established to enable swift remediation.

### Continuously assess the attack surface

New risks are consistently discovered in organizations' attack surfaces on a monthly basis. Continuous assessments is required in order to quickly find and remediate new exposures.

### Test web apps for injection risks

Injection risks accounted for nearly 60% of all critical severity risks discovered by Hadrian over the last 12 months. Organizations must continuously assess their applications for new vulnerabilities as they can be introduced at any time.

### Focus on limiting information leaks

Over 2/3 of application and service exposure risks have a low or informational severity score. While many threat actors utilize exposed RDP servers it is often with compromised credentials that they have obtained elsewhere. Therefore, the focus should be on limiting the amount of sensitive information that can be obtained by remediating Exposed Secrets, Injection Risks, and DNS Issues.

### Consider context when conducting severity scoring

Many of the most severe risks discovered by Hadrian do not have an associated CVE and the asset context is extremely impactful on the severity of a risk. As a result, CVSS base scores can not be relied upon for accurate prioritization. Implement stakeholder-specific vulnerability categorization methodologies to accurately prioritize risks.

### Improve collaboration workflows

Injection and misconfiguration risks, which are among the most likely to require support from developers and technology teams to remediate take the longest to be resolved at around 70 days which is nearly 3 times longer than easier to resolve issues. Clear communication of the urgency and action required is essential.

# Research Methodology

This report is based on Hadrian analysis of over 300 organizations' attack surfaces across multiple industries and regions between October 1, 2023, and October 1, 2024. The information was collected through active analysis to understand the exploitable exposures of each organization. Analysis was conducted across the entire attack surface including multiple cloud infrastructures, SaaS tools, and on-premise assets.

Only verified risks are analyzed in this report to remove false positives from the conclusions. Verified is defined by whether risk exposure is provable, this could be as simple as receiving specific or delayed responses from services to much more complicated methodologies.

The risk severity data categorized by severity—critical, high, medium, low, and informational— is based on Hadrian's proprietary context-based scoring system which has been built using machine learning techniques. The scoring includes factors such as business relevance and attractiveness of assets to attackers, along with discoverability, impact, and ease of exploitation. The results are periodically analyzed by Hadrian's in-house ethical hacking team to ensure accuracy.

Remediation effectiveness is assessed by tracking the average days between to discovery date and remediation. Hadrian automatically reassesses risks in order to validate that resolution is complete.

# About Hadrian

Gain complete control of your external attack surface by remediating exposures and hardening your attack surface. Hadrian is modernizing offensive security practices with automation, making security teams faster and more scalable. Continuously equipped with the hacker's perspective, companies make themselves hard to hack.

Hadrian provides companies with a real-time exposure management platform, viewing security through a hacker's eyes because, well, hackers understand hackers best. We continuously map the digital footprint of organizations, discover risks, and prioritize remediation for security teams to harden their external attack surface.

The Hadrian platform combines real-time asset discovery, continuous automated pen testing, and prioritization and remediation steps for exploitable vulnerabilities. The platform is agentless, quick to deploy and easy to use.

### Recognised by leading analysts

Hadrian is only vendor recognized as both a Leader and Outperformer in the 2024 GigaOm Radar Report for Attack Surface Management. Learn more at hadrian.io/gigaom.



> **"Hadrian's strengths are manifold, with its active assessment of vulnerabilities being a key highlight, thanks to its sophisticated Orchestrator AI"**
>
> Chris Ray
> Analyst at GigaOm

### Trusted worldwide by market leaders

| | | | | | |
|---|---|---|---|---|---|
| NBC | amadeus | BLINQX | CRÉDIT AGRICOLE | AUTODESK | SHV ENERGY |
| ABN·AMRO | London Business School | RITUALS... | SIEMENS energy | LOTTOMATICA | LEROY MERLIN |
| BIOLANDES | BLINQX | WeatherTech | =exact | nedap | Van Oord Marine ingenuity |