NETSPI™

OFFENSIVE SECURITY

VISION REPORT 2023

A look back – and forward – at vulnerability trends and patterns

HELLO & WELCOME!

Welcome to NetSPI's inaugural Offensive Security Vision Report! This data analysis examines the top vulnerability findings and trends from the past year in an effort to help security and business leaders focus discovery, management, and remediation efforts. Security teams today are seeing a rising number of vulnerabilities. According to the NIST National Vulnerability Database vulnerability count has steadily increased year-over-year for the past five years – and shows no signs of slowing down. This, coupled with the reality of burnt-out security and development teams, creates an imminent need for prioritization. While this report is packed with data, one narrative became clear during our analysis: vulnerability and asset discovery is table stakes. Today, offensive security is only as valuable as its ability to help you prioritize remediation of the issues that matter most to your business.

In this report, you'll find answers to the following questions, and more:

- Which [impactful] vulnerabilities are most pervasive across core application, cloud, and network attack surfaces?
- Which attack surfaces present the lowest and highest risk?
- Which industries hold the lowest and highest risk vulnerabilities?
- What are today's requirements for remediation efforts (SLAs)?
- What are the greatest barriers to timely and effective remediation?

NETSPI™

WHAT'S INSIDE?

Welcome **2-5**
- Methodology 4
- Glossary 5

TL;DR **6**

Top Vulnerabilities by Attack Surface **7-9**
- Severity Breakdown and Synopsis 8
- Industry Breakdown and Synopsis 9
- Overview of Remediation Tips 10

Applications **11-16**
- Web 12
- Mobile 13-14
- Thick Client 15-16

Cloud **17-19**

Network **20-26**
- External 21-23
- Internal 24-26

State of Remediation Survey **27-30**
- Remediation Due Dates 28
- Greatest Barriers 29

2023 Cybersecurity Hiring Trends **30**

Recommended Actions & Final Words **31**

About NetSPI **32-33**

Acknowledgements **34**

NETSPI

METHODOLOGY

We analyzed over 300,000 anonymized findings from thousands of pentest engagements spanning more than 240,000 hours of testing. Initially, we pulled the top 30 most prevalent vulnerabilities from our six core focus areas, or attack surfaces, from Resolve™, NetSPI's penetration testing as a service (PTaaS) platform.

The top 30 were determined by the following core variables:

- Only medium, high, and critical severities were reported.
- There were multiple instances of the finding across different company environments.
- The findings were exploitable on multiple occasions.

From the top 30 vulnerabilities, our offensive security experts were tasked with manually identifying 3-5 findings that security teams should prioritize this year. The vulnerabilities that follow are based on likelihood and impact.

While this inaugural edition of the Vision Report focuses on 2022 pentest data, in future reports you can expect a focus on observations from year over year analysis.

WE ANALYZED OVER
300,000
ANONYMIZED FINDINGS

PENTESTING ENGAGEMENTS INCLUDED MORE THAN
240,000
HOURS OF TESTING

WE PULLED THE
TOP 30
MOST PREVALENT VULNERABILITIES FROM OUR SIX CORE FOCUS AREAS

Sample Breakdown:

Industries:
Education
Energy & Utilities
Financial Services
Government & Non-Profit
Healthcare
Insurance
Manufacturing
Media & Entertainment
Professional Services
Real Estate
Retail & Ecommerce
Technology

Attack Surfaces:
Web Applications
Mobile Applications
Thick Applications
Cloud
External Network
Internal Network

Titles:
CISO
CSO
CTO
Director of IT
Director of Product Security
Global Head/Lead of Cybersecurity
Information Security Manager
VP of Information Security

NETSPI

GLOSSARY

Vulnerabilities:
Weaknesses in systems, applications, security procedures, or security controls that could be exploited by a malicious actor.

Findings:
A vulnerability that is tied to an asset, or group of assets.

Instances:
A single occurrence of a detected vulnerability on a particular asset.

Entry Points:
Vulnerabilities that are exploited to gain initial access to targeted systems, applications, or sensitive data.

SLAs/Remediation Due Dates:
The required timeframe to fix a vulnerability – from initial discovery to remediation.

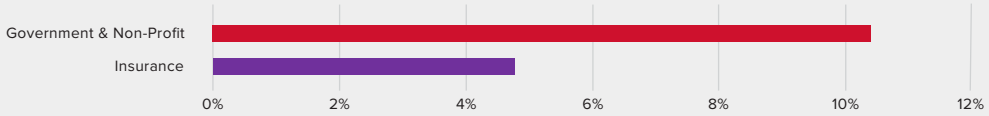
Severities, Defined.

Critical	Vulnerabilities that were successfully exploited during testing to gain initial access to targeted systems, applications, or sensitive data. *Ex: Code execution, arbitrary file read/write, PII/PHI/PCI data exposure, authorization/authentication control bypasses*
High	Vulnerabilities with the potential to provide direct, unauthorized access to protected networks, systems, application functionality, or sensitive data. *Ex: Code execution, arbitrary file read/write, PII/PHI/PCI data exposure, authorization/authentication control bypasses*
Medium	Vulnerabilities that result in the exposure of session data, security configuration information, unencrypted transmission of sensitive data, or use of weak encryption methods. This also includes vulnerabilities that require authentication, user interaction, or secure tier execution. *Ex: Cross-site scripting, open SMTP relay, clear text storage of passwords/sensitive data, and cleartext management protocols such as telnet*
Low	Vulnerabilities that disclose non-critical host information, non-critical weak cryptography configurations, and best practices that don't directly lead to unauthorized access to protected networks, systems, application functionality, or sensitive data. *Ex: SSL and RDP findings, password policy best practices, missing HTTP security headers, software version disclosures*

TOO LONG; DIDN'T READ

Don't have time to read the full report?
Here are a few findings *you won't want to miss.*

On average, the highest volume of critical and high severity vulnerabilities were discovered within the government and nonprofit industry. On the contrary, insurance had the lowest volume of critical and high severity vulnerabilities.



LACK OF RESOURCES AND PRIORITIZATION

are the two greatest barriers to timely and effective remediation today.

Of the applications tested, **WEB APPLICATIONS** have a higher prevalence of **HIGH AND CRITICAL VULNERABILITIES** compared to mobile and thick applications.

Internal networks have nearly

3x

more exploitable vulnerabilities than external networks.

71%

of respondents shared that less than one-fourth of roles budgeted were entry level, with

46%

of those reporting no plans for entry level hiring in 2023.

TOP VULNERABILITIES (BY ATTACK SURFACE) TO PRIORITIZE INCLUDE:

WEB APPLICATIONS

- Authorization Bypass – Missing Function Level Access Controls
- Authorization Bypass – Insecure Direct Object References
- SQL Injection

MOBILE APPLICATIONS

- Authorization Bypass – Insecure Direct Object References
- Authorization Bypass – Missing Function Level Access Controls
- Sensitive Information Disclosure – Mobile Application Files
- Authentication Bypass – Biometric

THICK APPLICATIONS

- Authorization Bypass – Missing Function Level Access Controls
- Authorization Bypass – Client-Side Controls
- Cleartext Protocol
- Insecure Architecture – Two-Tier
- Access Control Weakness – No Authentication

CLOUD

- Publicly Available Resources Hosting Sensitive Data
- Misconfigured or Permissive IAM Permissions
- Cleartext Credential Storage
- Vulnerable Software and OS Versions (Missing Critical Patches)

EXTERNAL NETWORK

- Publicly Available Resources Hosting Sensitive Data
- Weak or Default Passwords (Active Directory and Application Accounts)
- Remote Management Interface – Web Administration
- Web Application Vulnerabilities
- Vulnerable Software and OS Versions (Missing Critical Patches)

INTERNAL NETWORK

- Network Protocol Attacks
- Weak or Default Passwords (Active Directory and Application Accounts)
- Excessive Permissions
- Weak Configurations
- Missing Critical Patches
- Web Application Vulnerabilities

NETSPI™

TOP VULNERABILITIES

BY ATTACK SURFACE

We asked our global penetration testing experts to review the top 30 most prevalent medium, high, and critical severity vulnerabilities from thousands of tests across six attack surfaces. From those lists, they identified 3-5 findings that would pose the greatest risk if found in a client environment.

This report is intended to help security teams focus their discovery and remediation efforts on the riskiest vulnerabilities most likely to exist on their attack surface. Plus, we'll get you started on the right path toward remediation with best practices and recommendations from our global offensive security experts.

Before we dive into the individual findings, here is a snapshot of the findings for each attack surface, delineated by severity. From this analysis, we uncovered that, of the applications tested, web applications have a higher prevalence of high and critical vulnerabilities compared to mobile and thick applications, and overall.

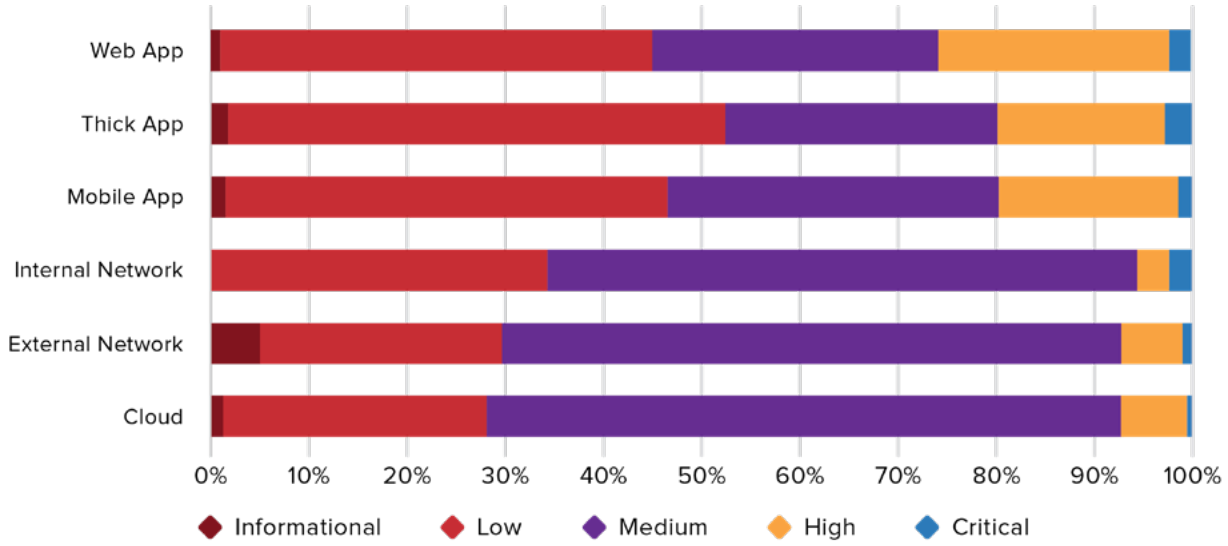
We also analyzed entry points, or vulnerabilities that were deemed exploitable, finding that internal networks have nearly 3x more exploitable vulnerabilities than external networks on average. Beyond the fact that the external attack surface is smaller than the internal network, the external network's lower exploitability could be due to the external attack surface remaining a higher priority for companies during remediation because it represents a higher risk due to its exposure to the internet.

Internal networks have nearly

3x

more exploitable vulnerabilities than external networks on average.

Breakdown of Severity, by Attack Surface

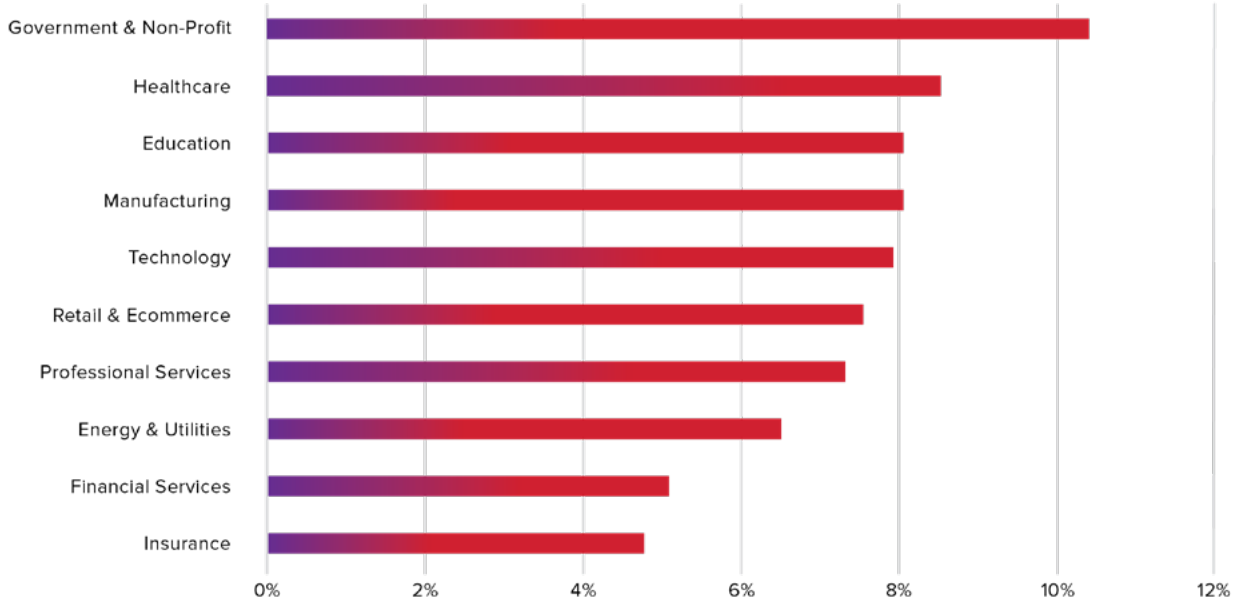


We also explored high-level industry data. On average, the largest volume of critical and high severity vulnerabilities were found within the government and non-profit and healthcare industries. On the contrary, insurance and financial services had the lowest volume of critical and high severity vulnerabilities.

Data shows two highly regulated industries at both ends of the spectrum: healthcare and financial services. In conversations held with healthcare security leaders, we've heard that it is no easy feat to keep up with privacy regulations. Perhaps the complexity of the healthcare regulatory environment is distracting it from focusing on risk-based security? It will be interesting to see how this data changes over the next few years. Financial services regulators have certainly shifted to evaluate and penalize risk management deficiencies – will we see healthcare follow this path? This may also be a signal that further collaboration and information sharing across industries is needed.

Energy and utilities is a curious top three industry with a lower percentage of severe vulnerabilities. Over the past two years there has been an emphasis on securing industrial control systems (ICS), notably from government agencies (EPA, CISA). Perhaps the awareness around the criticality of securing these systems is paying off?

Percentage of High and Critical Vulnerabilities, by Industry



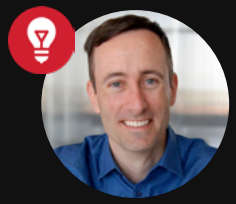
The highest volume of critical and high-severity vulnerabilities were discovered within the government and nonprofit industry.

NETSPI™

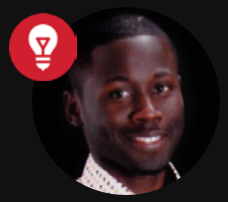
REMEDIATION TIPS

In the attack surface analysis sections that follow you'll find remediation tips straight from our global offensive security experts. Adopt these tips and tricks along your journey to effective remediation!

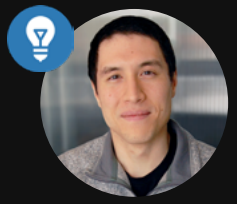
Paul Ryan
Director
Application Pentesting



Andy Acer
Director
Mobile Pentesting



Andre Joseph
Director
Thick Client Pentesting



Thomas Elling
Director
Cloud Pentesting

Ryan Krause
Principal Consultant
External Network Pentesting

Josh Weber
Director
Internal Network Pentesting

NETSPI™

APPLICATIONS

WEB, MOBILE, AND THICK CLIENT

Vulnerabilities in web, mobile, and thick client applications are a significant threat to organizations and their data. Applications, whether developed internally or acquired from an external source, often have a considerable attack surface, process sensitive data, perform business critical operations, and have a variety of user roles. It's this variety and complexity that makes applications a prime target for attackers and a core focus of NetSPI's penetration testing service lineup.

During analysis, we found a few key observations, including:

- Web applications have the highest volume of high and critical vulnerabilities of all types of tests that NetSPI performs. This is likely due to the high exposure of internet facing web applications and that there are more web application penetration tests performed than any other assessment. Not to mention web application pentesting is in NetSPI's DNA and our methodology is very collaborative to fully prove out vulnerabilities and demonstrate their full impact.

- Access Control issues are top findings for all three application penetration tests. While validating authorization prior to access of functionality or data differs depending on application technology, the importance of performing this check is crucial to the confidentiality, integrity, and availability of the data.

- Many of the vulnerabilities listed here require human-driven pentesting to discover. As applications and APIs become more complex, human intuition and understanding will become even more essential to root out security weaknesses in business logic, cross-application interactions, and authorization controls. NetSPI's testing methodology focuses on these complex and high risk issues to identify vulnerabilities that automated tools are unable to correctly identify.

WEB APPLICATIONS

Authorization Bypass – Missing Function Level Access Controls (MFLAC)

If an MFLAC vulnerability exists, the application does not perform adequate access control checks and unauthorized users can perform actions outside of their intended scope of permissions. This can result in the access, modification, or deletion of data within the system. In the most severe instances, it may be used for privilege escalation.

It is extremely prevalent in web applications and can be difficult to identify every instance of it. Given how severe it can be, it will be one of the likeliest attack paths to theft of data in a system.

Authorization Bypass – Insecure Direct Object References

IDOR vulnerabilities permit a user to view unauthorized data (objects) within a system based on user-controlled input. Typically, a user will have access to their own record but by modifying a parameter, a cookie, or other input, the attacker will then be able to view similar type records of other users. This is worth looking into due to the sensitivity of the unauthorized information that an attacker can retrieve. Depending on the application, attackers could view private personal data, health information, or financial records. Often, broken access control failures such as IDOR (and MFLAC) require a human-driven penetration test to discover by understanding how the application should work.

REMEDIATION TIP

For both MFLAC and IDOR Authorization Bypass findings, remediation best practices are similar. Fine-grained access controls should be implemented to properly attribute authorization of records/objects as well as functions to the individually authenticated and authorized user.

Paul Ryan
Director, Application Pentesting

SQL Injection

SQL injection occurs when unvalidated data is used in a dynamically built backend SQL query. An attacker injects data that is interpreted as part of the SQL query leading to unauthorized access to create, read, update, and delete data from the database. Depending on the configuration of the server, SQL injection can also lead to remote code execution.

Because an attacker can read, modify, and delete data from the database, SQL injection can affect all areas of CIA – confidentiality, integrity, and availability. Again, depending on configuration, this can lead to remote code execution and can be considered an entry point into environments.

REMEDIATION TIP

Prepared statements with parameterized queries are the best defense against SQL injection. Untrusted user input is first prepared as a string and then entered into the query rather than being concatenated directly into the query. This preparation of the parameters allows the database to distinguish between data and code.

Paul Ryan
Director, Application Pentesting

MOBILE APPLICATIONS

Authorization Bypasses – Insecure Direct Object References (IDOR) and Missing Function Level Access Controls (MFLAC)

Mobile applications can be susceptible to IDOR and MFLAC vulnerabilities in the same way as web applications. IDOR vulnerabilities are a privilege escalation flaw that allow one user to access another user's data.

Many mobile applications receive less scrutiny on their server-side APIs because there is greater technical complexity involved in performing these reviews. With an MFLAC vulnerability, the server does not restrict access to sensitive functionality. An attacker that identifies this can use the functionality maliciously. An example could involve administrative functionality such as a password reset, or account unlock. If such functionality were unrestricted, an attacker could hijack the accounts of other users.

These types of attacks are direct attack scenarios, meaning they are exploitable over the internet by an attacker. An attacker who becomes aware of these types of vulnerabilities can exploit them without needing to target a specific user or gain access to a user's device.

The consequences of exploitation can be significant in privilege escalation vulnerabilities like IDOR and MFLAC findings.

Sensitive Information Disclosure – Mobile Application Files

Sensitive information stored on mobile devices can be exposed in a Lost Device scenario. This can include Personal Identifiable Information (PII) of the user, authorization tokens, API keys, or sensitive application data.

If this vulnerability exists, an attacker who steals a targeted user's device could extract the data stored on it and gain access to the stored data. If sensitive, the data can be immediately beneficial to the attacker. Alternatively, access keys or tokens could be used to gain online access to additional resources. This could allow the attacker to impersonate the user in the context of the application, discover sensitive data, or identify the means of engaging in follow-on attacks.

REMEDIATION TIP

To remediate a Sensitive Information Disclosure finding, ensure that locally stored data is stored in an encrypted form by using encryption keys that are either stored server-side or locally in the Keystore or Keychain and protected by the Secure Element (SE), if available. For locally stored encryption keys, set strong use restrictions on their use to ensure that application data isn't automatically decrypted at startup and to ensure that the keys aren't exposed in application backups.

Andy Acer
Director, Mobile Pentesting

Many mobile applications receive less scrutiny on their server-side APIs because there is greater technical complexity involved in performing these reviews.

Authentication Bypass – Biometric

Biometric bypasses involve spoofing the application's client-side code into locally authenticating an attacker despite the attacker not presenting the correct biometric credentials. In this attack flow, bypassing the local authentication step leads to a fully authenticated session for the attacker.

An attacker who steals a targeted user's device could impersonate that user and use the application on their behalf. In a mobile banking example, the attacker could attempt to move funds from the targeted user's account to an account they control.

REMEDIATION TIP

When dealing with biometric bypasses, confirm stored authentication tokens are protected with encryption keys protected by the devices Secure Element (SE) and ensure strong protections are enforced for access requirements on the stored keys. Both Android and iOS have robust systems to protect authentication tokens using the Keystore or Keychain. By setting restrictions on key usage, the tokens will be protected even in a variety of attack scenarios.

Andy Acer
Director, Mobile Pentesting

Biometric bypasses involve spoofing the application's client-side code into locally authenticating an attacker despite the attacker not presenting the correct biometric credentials.

THICK CLIENT APPLICATIONS

Authorization Bypass – Missing Function Level Access Controls

Thick client applications can be susceptible to IDOR and MFLAC vulnerabilities in the same way as web and mobile applications. This showcases the pervasiveness of MFLAC vulnerabilities across the application ecosystem.

Authorization Bypass – Client-Side Controls

The server-side component of the application does not examine the data it retrieves from the client to validate if it is secure or correct. This vulnerability allows the client to perform unauthorized actions.

Thick, mobile, and embedded applications are more susceptible to this vulnerability than other kinds of applications because developers often do not consider the client to be untrusted. This is especially true for thick clients that are compiled directly to the platform's Instruction Set Architecture (ISA – e.g. x86_64, ARM, MIPS, etc.) and/or use statically compiled libraries responsible for data encoding and encryption. An attacker that has direct access to the client can eventually figure out how to build a malicious client and leverage this vulnerability to perform malicious activities.

REMEDIATION TIP

Ensure all client → server calls are checked for proper authorization on the server. Additionally, perform server-side input validation on the client → server call to ensure a malicious client cannot access functionality they aren't intended to access.

Andre Joseph
Director, Thick Client Pentesting

Thick, mobile, and embedded applications are more susceptible to this vulnerability than other kinds of applications because developers often do not consider the client to be untrusted.

Cleartext Protocol

Cleartext protocol occurs when the client does not encrypt the transport layer between itself and the server.

There is a plethora of "internal" applications that utilize this insecure method. Applications that usually fit this paradigm include billing software, medical software, and HR software. An attacker that managed to penetrate "external" controls could use this weakness to easily steal sensitive information.

REMEDIATION TIP

Ensure the transport layer between the client and server is encrypted with TLS 1.2+. Additionally, double check that the server does not support Cipher-Block-Chaining (CBC) as they are susceptible to padding oracle attacks.

Andre Joseph
Director, Thick Client Pentesting

Insecure Architecture – Two-Tier

Thick client applications directly connect to the backend database/datastore. This often allows a malicious user carte-blanche over the data in the data store. This insecure architecture is prevalent in many backend applications and usually exposes organizations to significant risks.

Similar to cleartext protocol vulnerabilities, there are many "internal" applications that utilize this insecure architecture. An attacker that manages to penetrate "external" controls could use this weakness to easily pivot into systems with sensitive information.

REMEDIATION TIP

Confirm there is an intermediate application (e.g. a web server) that can mediate information flow between the thick client and datastore/database. If this is not possible, ensure the application does not use one database account and provisions each user their own database/datastore account with Access Control Lists (ACLs) in place that only grant them access to data they should view/own.

Andre Joseph
Director, Thick Client Pentesting

Access Control Weakness – No Authentication

This vulnerability occurs when the server-side component of a thick client application does not check the data it retrieves from the client to see if it is authenticated. It allows the client to perform actions it is not authorized to perform.

An attacker that has direct access to the client can build a malicious client and perform exploits that could expose an organization to significant risks.

REMEDIATION TIP

Ensure that the client that issues a client → server call is properly authenticated for the call. Additionally, ensure proper input validation is performed on the client → server call to ensure a malicious client can't bypass authentication controls.

Andre Joseph
Director, Thick Client Pentesting

An attacker that has direct access to the client can build a malicious client and perform exploits that could expose an organization to significant risks.

NETSPI™

CLOUD

What is the common thread surrounding the top cloud vulnerabilities? The principle of least privilege must be applied to cloud security. NIST defines the principle as the idea “that a security architecture should be designed so that each entity is granted the minimum system resources and authorizations that the entity needs to perform its function.”

An attacker that gains access to an identity with excessive IAM permissions will have an abundance of options for data access, lateral movement, or privilege escalation. Likewise, locking down permissions to only what is needed can limit the blast radius in the event of compromise and limit privilege escalation vectors.

This principle is particularly important in the cloud as there are multiple layers of access controls that often have to be considered. IAM provided by the cloud provider can often influence permissions in other access control systems. Take for example a user that has been granted the most privileged access role in the cloud (Administrator Access/Owner). These managed roles often contain thousands of permissions, including full Administrative access over Virtual Machine workloads. If an Active Directory Domain Controller is running in the environment on a VM, the overly permissive Cloud IAM roles could inadvertently lead to users having Domain Administrator permissions in the Active Directory environment.

Publicly Available Resources Hosting Sensitive Data

A publicly available cloud resource allows public, anonymous access. This can apply to cloud services like storage or to IP addresses assigned to virtual machines.

Inadvertent public/anonymous access can lead to the exposure of sensitive data. In addition, this access could also potentially lead to privilege escalation vectors into the cloud environment.

REMEDIATION TIP

Ensure that all cloud services are restricted to internal, authenticated access if public access is not required. Employ a layered security approach that uses both individual service configuration settings and organization-wide policies as an additional guardrail.

Thomas Elling
Director, Cloud Pentesting

Misconfigured or Permissive IAM Permissions

This vulnerability indicates that a user, group, service account, or resource has been granted excessive IAM permissions beyond what they need to perform their daily duties.

Excessive IAM permissions violate the principle of least privilege and grant more permissions than necessary. This can create privilege escalation vectors or sensitive information disclosure in the environment.

REMEDIATION TIP

Follow the principle of least privilege and restrict IAM permissions to only permissions that are needed. Avoid the usage of overly permissive “basic” or “general” roles. Enforce additional security controls and just-in-time access to those identities that need privileged access.

Thomas Elling
Director, Cloud Pentesting

Excessive IAM permissions violate the principle of least privilege and grant more permissions than necessary.

Cleartext Credentials Storage

This finding means that a cleartext credential has been improperly stored in an environment variable, source code, or other cloud service that was not designed for secrets management.

Cleartext credential storage in places that were not intended for secrets management lack the protections of a dedicated secrets management service. Discovery of these cleartext credentials could lead to unauthorized access elsewhere in the environment. They could also be used for lateral movement and privilege escalation in the environment.

REMEDIATION TIP

Always store credentials in a service that is intended for secrets management. Ensure that access to the credentials follows the principle of least privilege.

Thomas Elling
Director, Cloud Pentesting

Cleartext credential storage in places that were not intended for secrets management lack the protections of a dedicated secrets management service.

Vulnerable Software and OS Versions (Missing Critical Patches)

This occurs when an unsupported or vulnerable software version was identified in the environment. This can be internally or externally exposed. Unsupported or vulnerable versions contain known security vulnerabilities that leave them open to exploitation.

While cloud providers aim to provide secure by default configurations and regular updates, the customer is still responsible for update management for services such as Virtual Machines. Vulnerable software versions with known security vulnerabilities can be an easy target for attackers while having high security impact. New exploits for common software are released frequently. Pervasive vulnerable versions can be indicative of improper update management or misconfiguration of cloud security features.

REMEDIATION TIP

Review the cloud provider's shared responsibility model to determine what is within the customer's responsibility for security. Ensure that regular patch management practices are in place to keep software updated. Upgrade all vulnerable and unsupported versions to supported versions that receive regular security updates.

Thomas Elling
Director, Cloud Pentesting

NETSPI™

NETWORK

With the increased availability of continuous external attack surface management (EASM) platforms, exposed or vulnerable assets and services are found more quickly than they have been in the past. Organizations tend to have well-defined patch management and security scanning practices for their external networks today, reducing the opportunity for attackers to take advantage of low-hanging fruit such as outdated software or misconfigured services. However, human error can still put organizations at risk and has become a key factor in some of the most common external network entry points.

Developers can accidentally commit company code that contains API keys or passwords to their personal GitHub repositories that are publicly searchable. System administrators can misconfigure a firewall rule that opens a management interface to unintended external IP addresses. Employees can select weak or easily guessable passwords that can be discovered during password spraying operations. Attackers can overwhelm an employee to the point of confusion via MFA bombing (sending many MFA requests to the target in quick succession), causing them to approve the request unintentionally.

While the occurrences of finding sensitive open ports or outdated software with one-click exploits on external testing engagements are getting much less common, the human factor continues to produce viable entry points and remains difficult for organizations to manage effectively.

Once a threat actor is able to establish a foothold internally, the attack surface increases significantly. Network protocol attacks could allow a malicious user to man-in-the-middle legitimate traffic to steal sensitive information or impersonate users. Weak passwords that could be limited by MFA externally may not have the same safeguards internally. Missing critical patches and web application vulnerabilities that are regularly tested and fixed externally may not have the same urgency internally, potentially allowing threat actors to exploit and move laterally within the environment. Excessive permissions and weak configurations could allow threat actors to access unintended internal resources, exposed credit card data, client PII, or even escalate their permissions to that of a highly privileged user – granting them complete control over the domain and all machines attached to it.

What is the moral of this story? Network security is paramount. Continue reading for key network security vulnerabilities to watch and prioritize remediation for.

EXTERNAL NETWORK

Publicly Available Resources Hosting Sensitive Data

Sensitive information such as credentials, API keys, and internal domain information can inadvertently be exposed in publicly accessible places such as online source code repositories, cloud storage platforms, and public paste sites.

Attackers may discover publicly accessible information and use it against the organization's employees and infrastructure. Credentials or API keys may allow an attacker to gain unauthorized access to an organization's systems or cloud services for example, while internal organizational details might be used to build effective pretext scenarios for targeted social engineering attacks.

Attackers may discover publicly accessible information and use it against the organization's employees and infrastructure.

REMEDIATION TIP

Ensure that effective policies, procedures, and monitoring solutions are established to safeguard the flow of organizational information to external locations. Review commonly targeted sources of information such as GitHub and Pastebin on a regular basis to identify and remove any sensitive information that may have been inadvertently disclosed.

Ryan Krause
Principal Consultant
External Network Pentesting

Weak or Default Passwords (Active Directory and Application Accounts)

This one may seem obvious, but the fact that it exists as a top external network vulnerability indicates there is still work to be done to improve password policies. Domain users that have weak or easily guessable passwords create a significant risk to data, applications, and systems.

Weak or guessable domain user credentials can be used to authenticate to exposed login portals that support domain-based authentication such as Azure AD, VPN portals, VDI portals, and other exposed web applications.

REMEDIATION TIP

Ensure strong policies are in place for domain passwords. Enforce longer minimum length requirements as well as account lockout thresholds to reduce an attacker's ability to easily guess passwords. Check passwords against lists of commonly used and breached passwords and prohibit their use. Implement multi-factor authentication throughout the organization to limit the impact of compromised credentials.

Ryan Krause
Principal Consultant
External Network Pentesting

Remote Management Interface – Web Administration

Externally exposed administrative web interfaces present an opportunity for an attacker to access sensitive functionality such as system configuration and user management that is normally restricted to authorized users within an organization.

Administrative portals can often become exposed unintentionally via incorrect or overly permissive firewall rules.
If a company has not changed the default password associated with a specific administrative interface, an attacker can potentially gain access to the application. Additionally, if user credentials are compromised and the administrative portal is accessible, an attacker can potentially use the compromised credentials to gain unauthorized access to the portal.
The portal may provide the attacker with access to sensitive information or enable the attacker to pivot further into the network by abusing administrative functionality.

Administrative portals can often become exposed unintentionally via incorrect or overly permissive firewall rules.

REMEDIATION TIP

Restrict external access to administrative interfaces using appropriate firewall rules and/or IP-based allow lists. If the interface isn't required to be externally accessible, restrict access such that it is only accessible from inside the network or over a VPN. When possible, require multi-factor authentication for any administrative users who access the management interface. Only permit access to individuals who have a required business need to access the remote management interface.

Ryan Krause
Principal Consultant
External Network Pentesting

Web Application Vulnerabilities

Web applications that are outdated, poorly built, or insecurely configured may contain a variety of vulnerabilities such as SQL injection and cross-site scripting. They may inadvertently expose sensitive information or functionality as a result of these vulnerabilities or misconfigurations.

Applications that are exposed to the internet represent a significant attack surface and can often contain severe injection-based vulnerabilities, authentication and authorization issues, and flaws that expose sensitive information to attackers. They often serve as entry points into an organization's network as well.

REMEDIATION TIP

Train application developers on secure coding practices and integrate foundational security best practices into the entire software development lifecycle. Keep components, libraries, and infrastructure used by the application patched. Ensure that proper threat modeling, static and dynamic application security testing, and manual penetration testing is performed on both your web applications and external network on a regular, or continuous basis.

Ryan Krause
Principal Consultant
External Network Pentesting

Vulnerable Software and OS Versions (Missing Critical Patches)

In this instance, a client runs unpatched software exposed to the internet that contains known security vulnerabilities, leaving them open to exploitation.

Unpatched software with known security vulnerabilities is an attractive target for attackers. New exploits are released on a regular basis by security researchers (as well as malicious hackers), and if left unpatched, outdated software can quickly become an entry point into the organization.

REMEDIATION TIP

Establish solid patch management practices to keep software updated. If a product version is no longer supported by the vendor, upgrade to a supported version that receives regular security patches. If no upgrades are available or the software is end-of-life, consider decommissioning the software or, at minimum, removing external access to it.

Ryan Krause
Principal Consultant, External Network Pentesting

INTERNAL NETWORK

Network Protocol Attacks

This vulnerability category includes most of the top network protocols that we frequently target to gain an initial foothold on an internal network, such as ARP, DHCPv6, LLMNR, MDNS, and NBNS. Most of these protocols are enabled by default and may be unknown or unused by the client organization.

Exploitation of these common protocols could allow an attacker to gain a man-in-the-middle position with unsuspecting users. This could lead to credential or sensitive data exposure, a foothold on the domain, and privilege escalation.

Exploitation of these common protocols could allow an attacker to gain a man-in-the-middle position with unsuspecting users.

REMEDIATION TIP

Remove support for commonly exploited protocols if they are not being utilized for a business purpose internally. For example, we frequently identify unutilized LLMNR and NBNS protocols unknowingly exposed on internal Windows networks, and disabling them through Group Policy could completely remove these attack vectors. Remediation steps will vary depending on the protocol exploited.

Josh Weber
Director
Internal Network Pentesting

Weak or Default Passwords (Active Directory and Application Accounts)

This finding focuses on user passwords that can be guessed (Ex: Season + year, CompanyName + year, 'Password1', default vendor passwords, etc.).

Weak or guessable passwords can be used to authenticate to interfaces that support domain-based authentication. Additionally, default credentials are easy to identify based on applications and versioning. Unlike interfaces exposed externally, we rarely see interfaces on the internal network supporting or requiring multi-factor authentication, making the likelihood of compromise much greater.

REMEDIATION TIP

Reiterating Ryan's suggested best practices above, ensure strong policies are in place for all passwords. Enforce longer minimum length requirements as well as account lockout thresholds to reduce an attacker's ability to easily guess passwords. Check passwords against lists of commonly used and breached passwords and prohibit their use.

Josh Weber
Director
Internal Network Pentesting

Excessive Permissions

This category covers excessive permissions granted to users or groups. For example, we frequently see network shares or SQL servers that allow all Domain Users access, usually unintentionally, that contain sensitive information, credentials to other services or applications, or potentially even customer data (credit card numbers, PII, etc.). Unexpected excessive privileges could lead to a large number of internal users having access to unintended sensitive data.

REMEDIATION TIP

Identify excessive permissions on network shares and limit where possible. To get started, NetSPI VP of Research Scott Sutherland recently released PowerHuntShares, which is designed to automatically inventory, analyze, and report excessive privilege assigned to SMB shares on Active Directory domain joined computers. Scott also developed PowerUpSQL to support SQL server discovery, weak configuration auditing, privilege escalation at scale, and post-exploitation actions. Both tools are incredibly powerful for identifying and remediating excessive permissions vulnerabilities.

Josh Weber
Director, Internal Network Pentesting

Weak Configurations

This vulnerability category covers common misconfigurations we are seeing in client environments. This includes weak server configurations, utilization of uncommon groups, and, thanks to the research of the SpectorOps group, we are also seeing frequent misconfigurations associated with Active Directory Certificate Services.

The common weak configurations could lead to elevation of privileges and lateral movement within the environment, with some of the worst-case scenarios allowing threat actors to abuse a misconfiguration to elevate directly to Domain Admin.

REMEDIATION TIP

Ensure system hardening and change management policies are followed. We often see misconfigurations associated with development or test servers on the network, which can get easily overlooked in a fast paced environment. Regular network penetration tests or threat hunting exercises (internal and/or third-party) can also be utilized to ensure the organization is staying up to date on the latest attack vectors.

Josh Weber
Director, Internal Network Pentesting

Missing critical patches (e.g. MS17-010) could lead to direct code execution on systems.

Missing Critical Patches

Missing critical patches (e.g. MS17-010) could lead to direct code execution on systems. Unpatched software with known security vulnerabilities is an attractive target for attackers.

New exploits are discovered on a regular basis by security researchers (as well as threat actors), and if left unpatched, outdated software can quickly become an entry point into the organization.

Follow the best practices outlined earlier by Ryan Krause under Vulnerable Software Versions. Given we're looking at the internal network, if upgrades are not available or the software is end of life, at minimum isolate the system with technical controls.

Web Application Vulnerabilities

Applications that are exposed to internal users represent an attack surface and can often contain severe injection-based vulnerabilities, authentication and authorization issues, and flaws that expose sensitive information to attackers. They can often serve as entry points into an organization's network as well.

Follow the remediation steps outlined by Ryan Krause for external network. Focus on improving secure coding practices and implement a continuous security testing strategy.

NETSPI™

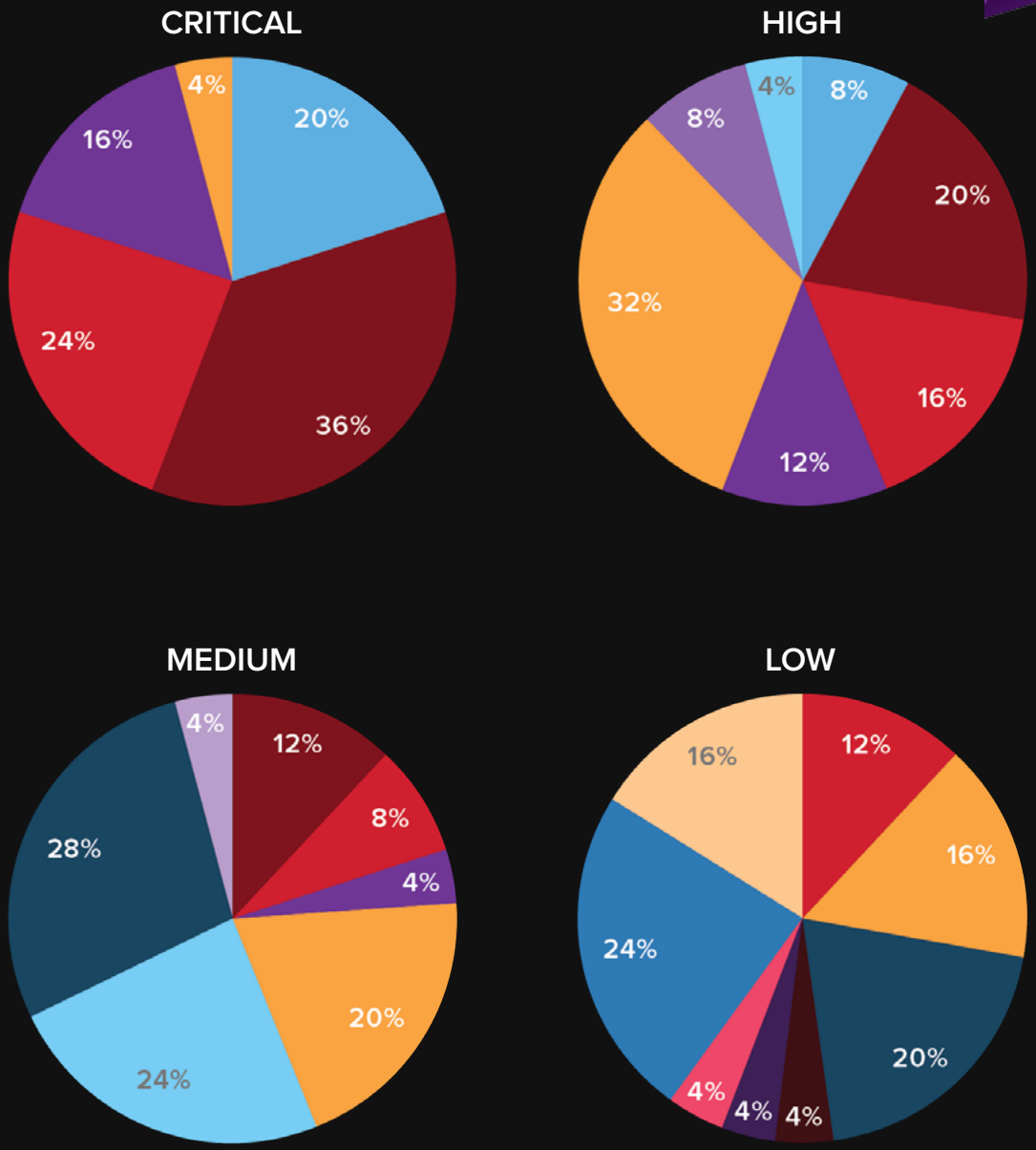
THE STATE OF REMEDIATION

We surveyed cybersecurity leaders across industries and regions to gain a better understanding of the current state of vulnerability remediation, tracking trends across due dates as well as the challenges facing timely and effective remediation today.

We acknowledge that remediation due dates and SLAs are greatly dependent on several variables, including industry, size and maturity of organization, risk tolerance, and regulations. However, we encourage this benchmark to prompt a review of your existing due dates.

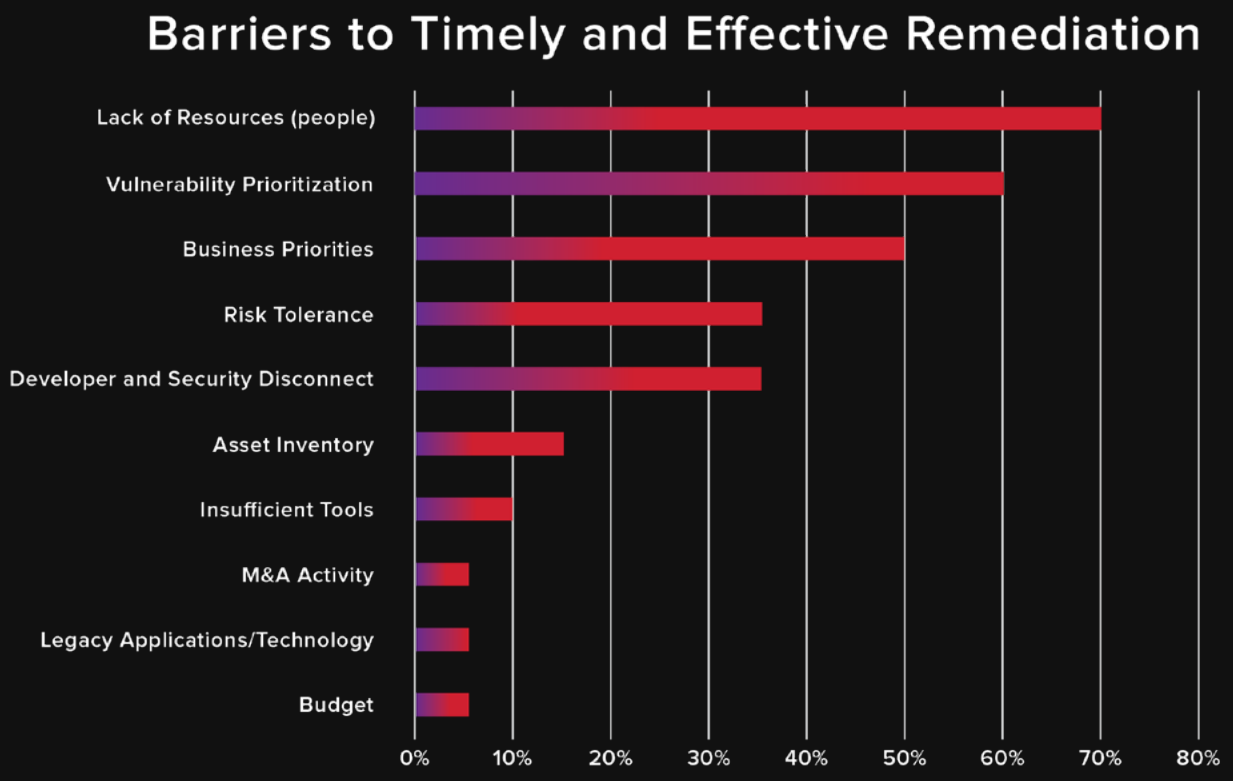
SEVERITY	AVERAGE	MAJORITY
Critical	6 days	1-6 days
High	20 days	28-34 days
Medium	54 days	84-90 days
Low	106 days	175-181 days

Remediation SLAs, by Severity



WHAT ARE THE GREATEST BARRIERS TO TIMELY AND EFFECTIVE REMEDIATION?

We asked security leaders to select their top three barriers to timely and effective remediation. Here's what they had to say:



Lack of resources, vulnerability prioritization, and business priorities were reported as the top three barriers to timely and effective remediation. The trend across all three? Security teams need support prioritizing the increasing number of vulnerabilities present in their environment.

We've all heard the adage, "you can't boil the ocean." If you take too much on all at once, it becomes difficult or even impossible to find a solution. This applies to vulnerability management. The reality is that we cannot fix every single vulnerability discovered – we must focus on which pose the greatest risk if exploited based on where they exist, what the business priorities are, the likelihood of exploitation, the threat landscape, among other factors.

Business and human context remains necessary to overcome vulnerability prioritization challenges, yet teams remain short staffed. Our call to the industry? Technology development over the next few years must support prioritization.

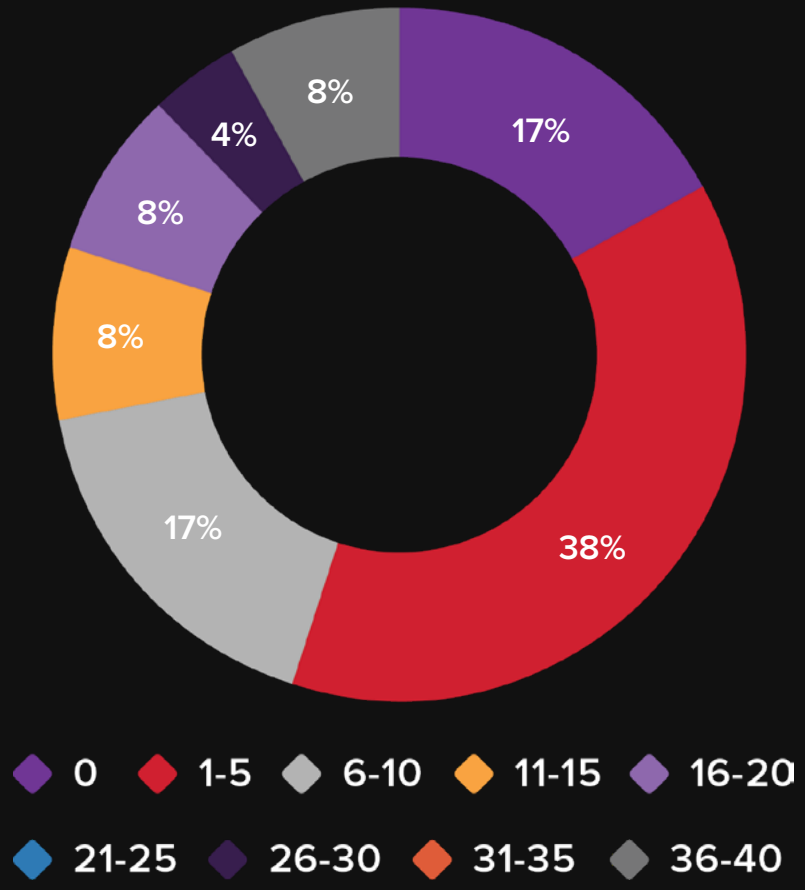
THE NEED FOR INCREASED INVESTMENT IN ENTRY LEVEL CYBERSECURITY ROLES

The majority (55%) of security leaders surveyed reported five or less roles budgeted for in 2023.

We followed this question up and asked how many of their roles were entry level. 71% of respondents shared that less than one-fourth of roles budgeted were entry level, with 46% reporting no plans for entry level hiring in 2023.

This indicates the need for further investment in new talent across the industry. We believe investment and hands on training for those new to the cyber industry is a key solution to the skills gap we are experiencing globally today.

Cyber Roles Budgeted for in 2023



RECOMMENDED ACTIONS & FINAL WORDS

Internally at NetSPI, we have the unique benefit of sharing vulnerability insights with one another to continuously improve our testing processes and keep pulse on the latest adversarial techniques. This report is a direct reflection of that wealth of knowledge that I get to see day in and day out – and I'm excited to have the opportunity to give other security and business leaders an opportunity to experience this for themselves.

If you've made it this far, it should be abundantly clear that there's much to be done to support and enable the industry to improve vulnerability prioritization. We hope the observations and actionable recommendations throughout our inaugural Offensive Security Vision Report are a solid data-driven starting point for you and your teams.

We can't wait to start on next year's report where we'll compare data year-over-year and dig into the data further to better understand the current state of offensive security and opportunities for advancement. For now, I'll leave you with a checklist of five immediate ways to incorporate this report in your security program:

- Work with your pentesting team to explore your attack surface for the findings listed within this report. Perform regular pentesting that demonstrates impact by exploitation of vulnerabilities. The most common vulnerabilities documented in this report would be priorities for remediation.
- Integrate security early and often – include security engineers and pentest teams as early as possible in the design and implementation process. It is much easier to integrate security early rather than retroactively. Consider partnering security engineers and pentesters with development teams.
- Review remediation due dates and, if needed, update with the State of Remediation survey results as a benchmark.
- Brainstorm creative opportunities to invest in entry level cybersecurity roles across your organization.
- Establish security best practices and guardrails after pentests – vulnerabilities should always be reviewed for root cause to avoid the same or similar vulnerabilities popping up in the future. Document remediation and make security awareness a priority after pentests.

Thanks for reading!

Aaron Shilts
CEO at NetSPI

COMPREHENSIVE VULNERABILITY DISCOVERY AND PRIORITIZATION

ASM

Discover and prioritize external attack surface exposures

RESOLVE

Discover and prioritize misconfigurations and vulnerability remediation

BAS

Discover and prioritize detective control improvements

> "When my NetSPI team has found something that is very high-risk or critical, they don't hesitate to let us know about it, lock it down, and approach the problem in a way that insulates us from being exploited until we can remediate it."
>
> **Phil Morris**
> Director, Enterprise AppSec and DevOps Governance, Healthcare Software

Get a demo today!

CONTACT US

NETSPI™

ABOUT NETSPI

NetSPI is the global leader in offensive security, delivering the most comprehensive suite of penetration testing, attack surface management, and breach and attack simulation solutions. Through a combination of technology innovation and human ingenuity NetSPI helps organizations discover, prioritize, and remediate security vulnerabilities. Its global cybersecurity experts are committed to securing the world's most prominent organizations, including nine of the top 10 U.S. banks, four of the top five leading global cloud providers, four of the five largest healthcare companies, three FAANG companies, seven of the top 10 U.S. retailers & e-commerce companies, and many of the Fortune 500. NetSPI is headquartered in Minneapolis, MN, with global offices across the U.S., Canada, the UK, and India.

Platform Driven, Human Delivered.

RESOLVE | BAS | ASM

- Application Pentesting
- Cloud Pentesting
- Network Pentesting
- IoT Pentesting
- Secure Code Review
- Strategic Advisory
- Red Team Testing
- Social Engineering
- Blockchain Pentesting

NETSPI™

ACKNOWLEDGEMENTS

Tying back to the ethos of this report, the combination of technology and people, we wanted to extend a special “thank you” to all who supported and contributed to this report.

Including but not limited to:

Andy Acer
Steven Carter
Cody Chamberlain
James Dinh
Thomas Elling
Karl Fosaaen
Josh Johnson
Andre Joseph
Tori Judd
Dane Knudsen
Ryan Kollmann
Ryan Krause
Nick Landers
Dylan Murphy-Mancini
Tori Norris
Paul Ryan
Nicholas Stang
Scott Sutherland
Juli Thomas
Josh Weber