September 25, 2025

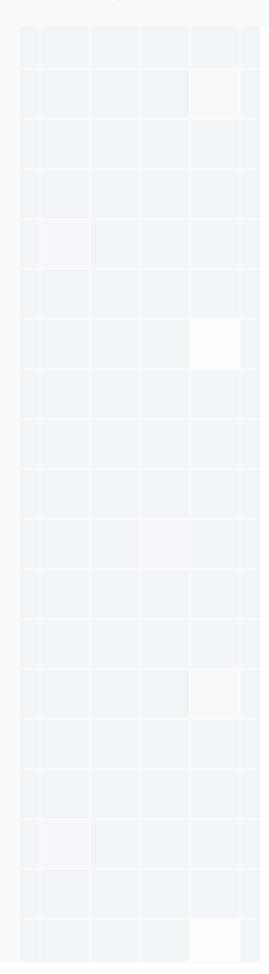
# Vulnerability Scan Report

Prepared By

**HostedScan Security** 



HostedScan Security Vulnerability Scan Report



### Overview

1 Executive Summary	3
2 Trends	4
3 Vulnerabilities By Target	5
4 Active Web Application Vulnerabilities	12
5 Passive Web Application Vulnerabilities	13
6 SSL/TLS Security	37
7 Network Vulnerabilities	38
8 Open TCP Ports	40
9 Open UDP Ports	56
10 Nuclei Vulnerabilities	57
11 Glossary	58



### **Executive Summary**

Vulnerability scans were conducted on select servers, networks, websites, and applications. This report contains the discovered potential vulnerabilities from these scans. Vulnerabilities have been classified by severity. Higher severity indicates a greater risk of a data breach, loss of integrity, or availability of the targets.

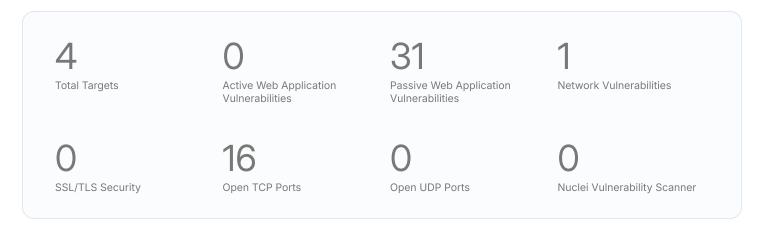
#### **Total Vulnerabilities**

Below are the total number of vulnerabilities found by severity. Critical vulnerabilities are the most severe and should be evaluated first. An accepted vulnerability is one which has been manually reviewed and classified as acceptable to not fix at this time, such as a false positive detection or an intentional part of the system's architecture.



#### **Report Coverage**

This report includes findings for **4 targets** scanned. Each target is a single URL, IP address, or fully qualified domain name (FQDN).



Vulnerability Scan Report

### **Trends**

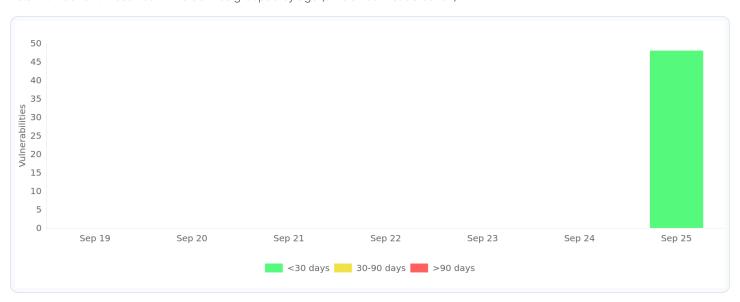
### **Open Risks**

Total number of vulnerabilities grouped by severity level.



### **Exposure Window**

Total number of unresolved vulnerabilities grouped by age (time since first detection).



Vulnerability Scan Report

### **Vulnerabilities By Target**

This section contains the vulnerability findings for each scanned target. Prioritization should be given to the targets with the highest severity vulnerabilities. However, it is important to take into account the purpose of each system and consider the potential impact a breach or an outage would have for the particular target.

### **Targets Summary** (4)

The number of potential vulnerabilities found for each target by severity.



### **Target Breakdowns**

Details for the potential vulnerabilities found for each target by scan type.



### krct.ac.in

#### **Total Risks**

0 0	22	10	0
69%			31%
Passive Web Application Vulnerabilities	Severity	First Detected	Last Detected
Absence of Anti-CSRF Tokens	Medium	0 days ago	0 days ago
CSP: Failure to Define Directive with No Fallback	Medium	0 days ago	0 days ago
CSP: Wildcard Directive	Medium	0 days ago	0 days ago
CSP: script-src unsafe-inline	Medium	0 days ago	0 days ago
CSP: style-src unsafe-inline	Medium	0 days ago	0 days ago
Content Security Policy (CSP) Header Not Set	Medium	0 days ago	0 days ago
Secure Pages Include Mixed Content (Including Scripts)	Medium	0 days ago	0 days ago
Cross-Domain Misconfiguration	Medium	0 days ago	0 days ago
Missing Anti-clickjacking Header	Medium	0 days ago	0 days ago
Vulnerable JS Library: bootstrap 4.0.0	Medium	0 days ago	0 days ago
Private IP Disclosure	Low	0 days ago	0 days ago
Cookie No HttpOnly Flag	Low	0 days ago	0 days ago
Cross-Domain JavaScript Source File Inclusion	Low	0 days ago	0 days ago
Cookie without SameSite Attribute	Low	0 days ago	0 days ago
Strict-Transport-Security Header Not Set	Low	0 days ago	0 days ago
X-Content-Type-Options Header Missing	Low	0 days ago	0 days ago
Cookie with SameSite Attribute None	Low	0 days ago	0 days ago

Server Leaks Version Information via "Server" HTTP Response Header Field	Low	0 days ago	0 days ago
Open TCP Ports	Severity	First Detected	Last Detected
Open TCP Port: 21	Medium	0 days ago	0 days ago
Open TCP Port: 22	Medium	0 days ago	0 days ago
Open TCP Port: 25	Medium	0 days ago	0 days ago
Open TCP Port: 26	Medium	0 days ago	0 days ago
Open TCP Port: 53	Medium	0 days ago	0 days ago
Open TCP Port: 110	Medium	0 days ago	0 days ago
Open TCP Port: 143	Medium	0 days ago	0 days ago
Open TCP Port: 465	Medium	0 days ago	0 days ago
Open TCP Port: 587	Medium	0 days ago	0 days ago
Open TCP Port: 993	Medium	0 days ago	0 days ago
Open TCP Port: 995	Medium	0 days ago	0 days ago
Open TCP Port: 3306	Medium	0 days ago	0 days ago
Open TCP Port: 80	Low	0 days ago	0 days ago
Open TCP Port: 443	Low	0 days ago	0 days ago
•	Low	0 days ago	0 days ago



### https://www.howtogeek.com/bypass-chromes-failedvirus-detected-message/

#### **Total Risks**

0 0	6	10	0
38%		63%	
Passive Web Application Vulnerabilities	Severity	First Detected	Last Detected
Absence of Anti-CSRF Tokens	Medium	0 days ago	0 days ago
CSP: Failure to Define Directive with No Fallback	Medium	0 days ago	0 days ago
CSP: Wildcard Directive	Medium	0 days ago	0 days ago
CSP: script-src unsafe-inline	Medium	0 days ago	0 days ago
CSP: style-src unsafe-inline	Medium	0 days ago	0 days ago
Content Security Policy (CSP) Header Not Set	Medium	0 days ago	0 days ago
Private IP Disclosure	Low	0 days ago	0 days ago
Cookie No HttpOnly Flag	Low	0 days ago	0 days ago
Cross-Domain JavaScript Source File Inclusion	Low	0 days ago	0 days ago
Cookie without SameSite Attribute	Low	0 days ago	0 days ago
Strict-Transport-Security Header Not Set	Low	0 days ago	0 days ago
Cookie Without Secure Flag	Low	0 days ago	0 days ago
Big Redirect Detected (Potential Sensitive Information Leak)	Low	0 days ago	0 days ago
Network Vulnerabilities	Severity	First Detected	Last Detected
TCP Timestamps Information Disclosure cvss score: 2.6	Low	0 days ago	0 days ago
Open TCP Ports	Severity	First Detected	Last Detected

Open TCP Port: 80	Low	0 days ago	0 days ago
Open TCP Port: 443	Low	0 days ago	0 days ago



### 152.57.199.249

Total Risks						
0						
		No vuli	nerabilities	s found.		



### https://krct.ac.in/ktgamin/assets/php/gallary/

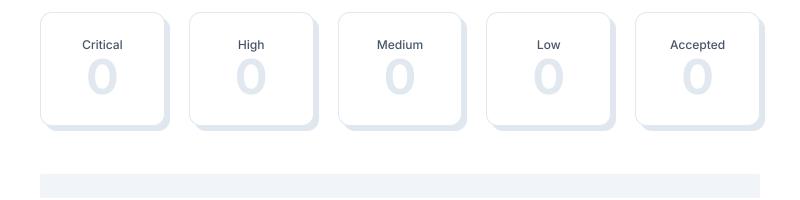
Total Risks						
		No vuli	nerabilities	found.		

### **Active Web Application Vulnerabilities**

The OWASP ZAP Active Web Application scan crawls the pages of a website or web application testing for vulnerabilities and configuration weaknesses. The active scan includes all of the passive scan tests and additionally makes requests and submits forms to actively test an application for more vulnerabilities. The active scan tests for vulnerabilities such as SQL injection, remote command execution, XSS, and more.

#### **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



#### **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

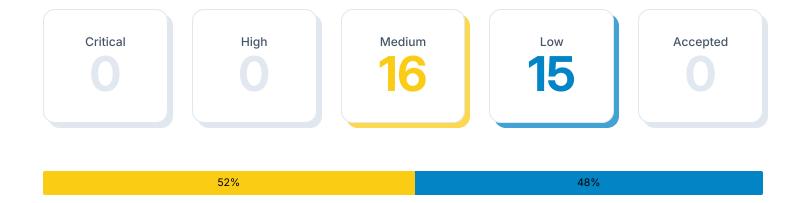
Title	Severity	Open	Accepted
No vulnerabilities detected			

### Passive Web Application Vulnerabilities

The OWASP ZAP Passive Web Application scan crawls the pages of a website or web application. The passive scan inspects each page as well as the requests and responses sent between the server. The passive scan checks for vulnerabilities such as cross-domain misconfigurations, insecure cookies, vulnerable Javascript dependencies, and more.

#### **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



#### **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

Title	Severity	Open	Accepted
Absence of Anti-CSRF Tokens	Medium	2	0
CSP: Failure to Define Directive with No Fallback	Medium	2	0
CSP: Wildcard Directive	Medium	2	0
CSP: script-src unsafe-inline	Medium	2	0
CSP: style-src unsafe-inline	Medium	2	0
Content Security Policy (CSP) Header Not Set	Medium	2	0
Secure Pages Include Mixed Content (Including Scripts)	Medium	1	0
Cross-Domain Misconfiguration	Medium	1	0
Missing Anti-clickjacking Header	Medium	1	0

Vulnerable JS Library: bootstrap 4.0.0	Medium	1	0
Private IP Disclosure	Low	2	0
Cookie No HttpOnly Flag	Low	2	0
Cross-Domain JavaScript Source File Inclusion	Low	2	0
Cookie without SameSite Attribute	Low	2	0
Strict-Transport-Security Header Not Set	Low	2	0
X-Content-Type-Options Header Missing	Low	1	0
Cookie with SameSite Attribute None	Low	1	0
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	1	0
Cookie Without Secure Flag	Low	1	0
Big Redirect Detected (Potential Sensitive Information Leak)	Low	1	0

### **Vulnerability Details**

Detailed information about each potential vulnerability found by the scan.



### Absence of Anti-CSRF Tokens

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

No Anti-CSRF tokens were found in a HTML submission form.

A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf.

CSRF attacks are effective in a number of situations, including:

- \* The victim has an active session on the target site.
- \* The victim is authenticated via HTTP auth on the target site.
- \* The victim is on the same local network as the target site.

CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.

#### Solution

Phase: Architecture and Design

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

For example, use anti-CSRF packages such as the OWASP CSRFGuard.

Phase: Implementation

Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.

Phase: Architecture and Design

Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330).

Note that this can be bypassed using XSS.

Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation.

Note that this can be bypassed using XSS.

Use the ESAPI Session Management control.

This control includes a component for CSRF.

Do not use the GET method for any request that triggers a state change.

Phase: Implementation

Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.

#### Instances (1 of 83)

uri: https://krct.ac.in/blog/2024/02/14/krct-offering-top-notch-training-and-engineering-placement-opportunities-in-trichy/method: GET

evidence: <form action="https://krct.ac.in/blog/wp-comments-post.php" method="post" id="commentform" class="comment-form"> otherinfo: No known Anti-CSRF token [anticsrf, CSRFToken, \_\_RequestVerificationToken, csrfmiddlewaretoken, authenticity\_token, OWASP\_CSRFTOKEN, anoncsrf, csrf\_token, \_csrf\_token, \_csrf\_token, \_csrf\_token, \_csrf\_token, \_csrf\_token, \_csrf\_token, \_csrfToken, data[\_Token][key], \_wpnonce] was found in the following HTML form: [Form 2: "author" "comment-submit" "comment\_parent" "comment\_post\_ID" "email" "url" "wp-comment-cookies-consent" ].

#### References

 $https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site\_Request\_Forgery\_Prevention\_Cheat\_Sheet.html \\ https://cwe.mitre.org/data/definitions/352.html$ 

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### CSP: Failure to Define Directive with No Fallback

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

The Content Security Policy fails to define one of the directives that has no fallback. Missing/excluding them is the same as allowing anything.

#### Solution

Ensure that your web server, application server, load balancer, etc. is properly configured to set the Content-Security-Policy header.

#### Instances (1 of 11)

uri: https://alumni.krct.ac.in/

method: GET

param: Content-Security-Policy evidence: frame-ancestors 'self'

otherinfo: The directive(s): form-action is/are among the directives that do not fallback to default-src.

#### References

https://www.w3.org/TR/CSP/

https://caniuse.com/#search=content+security+policy

https://content-security-policy.com/

https://github.com/HtmlUnit/htmlunit-csp

https://web.dev/articles/csp#resource-options

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### **CSP: Wildcard Directive**

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks. Including (but not limited to) Cross Site Scripting (XSS), and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

#### Solution

Ensure that your web server, application server, load balancer, etc. is properly configured to set the Content-Security-Policy header.

#### Instances (1 of 11)

uri: https://alumni.krct.ac.in/

method: GET

param: Content-Security-Policy evidence: frame-ancestors 'self'

otherinfo: The following directives either allow wildcard sources (or ancestors), are not defined, or are overly broadly defined: script-src, style-src, img-src, connect-src, frame-src, font-src, media-src, object-src, manifest-src, worker-src

#### References

https://www.w3.org/TR/CSP/

https://caniuse.com/#search=content+security+policy

https://content-security-policy.com/

https://github.com/HtmlUnit/htmlunit-csp

https://web.dev/articles/csp#resource-options

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### CSP: script-src unsafe-inline

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks. Including (but not limited to) Cross Site Scripting (XSS), and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

#### Solution

Ensure that your web server, application server, load balancer, etc. is properly configured to set the Content-Security-Policy header.

#### Instances (1 of 11)

uri: https://alumni.krct.ac.in/

method: GET

param: Content-Security-Policy evidence: frame-ancestors 'self'

otherinfo: script-src includes unsafe-inline.

#### References

https://www.w3.org/TR/CSP/

https://caniuse.com/#search=content+security+policy

https://content-security-policy.com/

https://github.com/HtmlUnit/htmlunit-csp

https://web.dev/articles/csp#resource-options

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### CSP: style-src unsafe-inline

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks. Including (but not limited to) Cross Site Scripting (XSS), and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

#### Solution

Ensure that your web server, application server, load balancer, etc. is properly configured to set the Content-Security-Policy header.

#### Instances (1 of 11)

uri: https://alumni.krct.ac.in/

method: GET

param: Content-Security-Policy evidence: frame-ancestors 'self'

otherinfo: style-src includes unsafe-inline.

#### References

https://www.w3.org/TR/CSP/

https://caniuse.com/#search=content+security+policy

https://content-security-policy.com/

https://github.com/HtmlUnit/htmlunit-csp

https://web.dev/articles/csp#resource-options

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Content Security Policy (CSP) Header Not Set

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Medium

2 targets

0 days ago

#### **Description**

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

#### Solution

Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

#### Instances (1 of 100)

uri: https://krct.ac.in/ method: GET

#### References

https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/CSP

https://cheatsheetseries.owasp.org/cheatsheets/Content\_Security\_Policy\_Cheat\_Sheet.html

https://www.w3.org/TR/CSP/

https://w3c.github.io/webappsec-csp/

https://web.dev/articles/csp

https://caniuse.com/#feat=contentsecuritypolicy

https://content-security-policy.com/

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Secure Pages Include Mixed Content (Including Scripts)

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

1 target

0 days ago

#### **Description**

The page includes mixed content, that is content accessed via HTTP instead of HTTPS.

#### Solution

A page that is available over SSL/TLS must be comprised completely of content which is transmitted over SSL/TLS.

The page must not contain any content that is transmitted over unencrypted HTTP.

This includes content from third party sites.

#### Instances (1 of 23)

uri: https://krct.ac.in/%09ktgadmin/assets/php/pdf/1574845012.pdf

method: GET

evidence: http://cdn.jsinit.directfwd.com/sk-jspark\_init.php

otherinfo: tag=script src=http://cdn.jsinit.directfwd.com/sk-jspark\_init.php

#### References

https://cheatsheetseries.owasp.org/cheatsheets/Transport\_Layer\_Protection\_Cheat\_Sheet.html

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Cross-Domain Misconfiguration

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

1 target

0 days ago

#### **Description**

Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server.

#### Solution

Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance).

Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.

#### Instances (1 of 1)

uri: https://alumni.krct.ac.in/img/favicon.png

method: GET

evidence: Access-Control-Allow-Origin: \*

otherinfo: The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.

#### References

https://vulncat.fortify.com/en/detail? category = HTML5 & subcategory = Overly % 20 Permissive % 20 CORS % 20 Policy + Property % 20 Permissive % 20 Policy + Property % 20 Permissive % 20

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Missing Anti-clickjacking Header

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Medium

1 target

0 days ago

#### **Description**

The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.

#### Solution

Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app.

If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.

#### Instances (1 of 100)

uri: https://krct.ac.in/ method: GET param: x-frame-options

#### References

https://developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Headers/X-Frame-Options

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Vulnerable JS Library: bootstrap 4.0.0

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Medium

1 target

0 days ago

#### **Description**

The identified library appears to be vulnerable.

#### Solution

Upgrade to the latest version of the affected library.

#### Instances (1 of 2)

uri: https://krct.ac.in/assets/bootstrap/js/bootstrap.min.js

method: GET

evidence: \* Bootstrap v4.0.0

otherinfo: The identified library bootstrap, version 4.0.0 is vulnerable. CVE-2018-14041 CVE-2019-8331 CVE-2018-14040 CVE-2018-14042 CVE-2024-6531 https://github.com/twbs/bootstrap/issues/28236 https://github.com/advisories/GHSA-pj7m-g53m-7638 https://www.herodevs.com/vulnerability-directory/cve-2024-6531 https://github.com/twbs/bootstrap/issues/20184 https://github.com/advisories/GHSA-vc8w-jr9v-vj7f https://nvd.nist.gov/vuln/detail/CVE-2024-6531 https://github.com/rubysec/ruby-advisory-db/blob/master/gems/bootstrap/CVE-2024-6531.yml https://github.com/twbs/bootstrap https://github.com/advisories/GHSA-9v3m-8fp8-mj99

#### References

https://owasp.org/Top10/A06\_2021-Vulnerable\_and\_Outdated\_Components/

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Private IP Disclosure

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

2 targets

0 days ago

#### **Description**

A private IP (such as 10.x.x.x, 172.x.x.x, 192.168.x.x) or an Amazon EC2 private hostname (for example, ip-10-0-56-78) has been found in the HTTP response body. This information might be helpful for further attacks targeting internal systems.

#### Solution

Remove the private IP address from the HTTP response body. For comments, use JSP/ASP/PHP comment instead of HTML/JavaScript comment which can be seen by client browsers.

#### Instances (1 of 2)

uri: https://krct.ac.in/about.php?cat=1&id=57

method: GET

evidence: 10.0.11.100

otherinfo: 10.0.11.100 192.168.61.61 192.168.49.115 192.168.49.115 192.168.49.115

https://datatracker.ietf.org/doc/html/rfc1918

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Cookie No HttpOnly Flag

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Low

2 targets

0 days ago

#### **Description**

A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.

#### Solution

Ensure that the HttpOnly flag is set for all cookies.

#### Instances (1 of 5)

uri: https://alumni.krct.ac.in/ method: GET

param: tz

evidence: Set-Cookie: tz

#### References

https://owasp.org/www-community/HttpOnly

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Cross-Domain JavaScript Source File Inclusion

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Low

2 targets

0 days ago

#### **Description**

The page includes one or more script files from a third-party domain.

#### **Solution**

Ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end users of the application.

#### Instances (1 of 100)

uri: https://krct.ac.in/

method: GET

param: https://app.fastbots.ai/embed.js

evidence: <script defer src="https://app.fastbots.ai/embed.js" data-bot-id="cm89uq7pm1vfirik9qce0cfmm"></script>

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Cookie without SameSite Attribute

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

2 targets

0 days ago

#### **Description**

A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.

#### Solution

Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.

#### Instances (1 of 3)

uri: https://krct.ac.in/blog/wp-comments-post.php

method: POST

param: comment\_author\_e95beb3e78fea43cd2571b6718db58a7

evidence: Set-Cookie: comment\_author\_e95beb3e78fea43cd2571b6718db58a7

#### References

https://datatracker.ietf.org/doc/html/draft-ietf-httpbis-cookie-same-site

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### Strict-Transport-Security Header Not Set

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

2 targets

0 days ago

#### **Description**

HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.

#### Solution

Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

#### Instances (1 of 100)

uri: https://krct.ac.in/ method: GET

#### References

https://cheatsheetseries.owasp.org/cheatsheets/HTTP\_Strict\_Transport\_Security\_Cheat\_Sheet.html

https://owasp.org/www-community/Security\_Headers

https://en.wikipedia.org/wiki/HTTP\_Strict\_Transport\_Security

https://caniuse.com/stricttransportsecurity

https://datatracker.ietf.org/doc/html/rfc6797

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



### X-Content-Type-Options Header Missing

SEVERITY

AFFECTED TARGETS

LAST DETECTED

Low

1 target

0 days ago

#### **Description**

The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.

#### Solution

Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages.

If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application/web server to not perform MIME-sniffing.

#### Instances (1 of 100)

uri: http://krct.ac.in/ktgadmin/assets/php/pdf/1595607815.pdf

method: GET

param: x-content-type-options

otherinfo: This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.

#### References

https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/compatibility/gg622941(v=vs.85) https://owasp.org/www-community/Security\_Headers

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Cookie with SameSite Attribute None

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

1 target

0 days ago

#### **Description**

A cookie has been set with its SameSite attribute set to "none", which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.

#### Solution

Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.

#### Instances (1 of 3)

uri: https://alumni.krct.ac.in/

method: GET param: encToken

evidence: Set-Cookie: encToken

#### References

https://datatracker.ietf.org/doc/html/draft-ietf-httpbis-cookie-same-site

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Server Leaks Version Information via "Server" HTTP Response Header Field

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

1 target

0 days ago

#### **Description**

The web/application server is leaking version information via the "Server" HTTP response header. Access to such information may facilitate attackers identifying other vulnerabilities your web/application server is subject to.

#### Solution

Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.

#### Instances (1 of 100)

uri: http://krct.ac.in/about.php?cat=1&id=90 method: GET evidence: nginx/1.25.5

#### References

https://httpd.apache.org/docs/current/mod/core.html#servertokens https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648552(v=pandp.10) https://www.troyhunt.com/shhh-dont-let-your-response-headers/

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



### Cookie Without Secure Flag

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

Low

1 target

0 days ago

#### **Description**

A cookie has been set without the secure flag, which means that the cookie can be accessed via unencrypted connections.

#### Solution

Whenever a cookie contains sensitive information or is a session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.

#### Instances (1 of 100)

uri: https://www.howtogeek.com/

method: GET

param: unauthenicatedArticleLimitReached

evidence: Set-Cookie: unauthenicatedArticleLimitReached

#### References

 $https://owasp.org/www-project-web-security-testing-guide/v41/4-Web\_Application\_Security\_Testing/06-Session\_Management\_Testing/02-Testing\_for\_Cookies\_Attributes.html$ 

Vulnerable Target	First Detected	Last Detected
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



## Big Redirect Detected (Potential Sensitive Information Leak)

SEVERITY AFFECTED TARGETS LAST DETECTED

Low 1 target 0 days ago

#### **Description**

The server has responded with a redirect that seems to provide a large response. This may indicate that although the server sent a redirect it also responded with body content (which may include sensitive details, PII, etc.).

#### Solution

Ensure that no sensitive information is leaked via redirect responses. Redirect responses should have almost no content.

#### Instances (1 of 2)

uri: https://www.howtogeek.com/promo/next/mural

method: GET

otherinfo: Location header URI length: 43 [https://www.howtogeek.com/promo/next/mural/]. Predicted response size: 343. Response Body Length: 418.

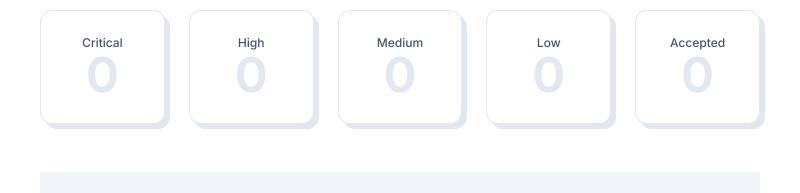
Vulnerable Target	First Detected	Last Detected
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago

# SSL/TLS Security

The SSLyze security scan tests for misconfigured SSL/TLS certificates, expired certificates, weak ciphers, and SSL/TLS vulnerabilities such as Heartbleed.

## **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



## **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

Title	Severity	Open Accepted
No vulnerabilities detected		

## **Network Vulnerabilities**

The OpenVAS network vulnerability scan tests servers and internet connected devices for over 150,000 vulnerabilities. OpenVAS uses the Common Vulnerability Scoring System (CVSS) to quantify the severity of findings. 0.0 is the lowest severity and 10.0 is the highest.

#### Lite Scan

Free accounts use the lite network scan which is limited to the 10 most common ports and excludes brute force tests.

### **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



100%

## **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.



## **Vulnerability Details**

Detailed information about each potential vulnerability found by the scan.



# TCP Timestamps Information Disclosure

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

**CVSS SCORE** 

PORT

Low

1 target

0 days ago

2.6

general/tcp

#### **Description**

The remote host implements TCP timestamps and therefore allows to compute the uptime.

The remote host implements TCP timestamps, as defined by RFC1323/RFC7323.

A side effect of this feature is that the uptime of the remote host can sometimes be computed.

It was detected that the host implements RFC1323/RFC7323.

The following timestamps were retrieved with a delay of 1 seconds in-between:

Packet 1: 254438574 Packet 2: 254439619

#### Solution

To disable TCP timestamps on linux add the line 'net.ipv4.tcp\_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.

To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled'

Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.

The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.

See the references for more information.

#### References

https://datatracker.ietf.org/doc/html/rfc1323

https://datatracker.ietf.org/doc/html/rfc7323

https://web.archive.org/web/20151213072445/http://www.microsoft.com/en-us/download/details.aspx?id=9152

https://www.fortiguard.com/psirt/FG-IR-16-090

Vulnerable Target	First Detected	Last Detected
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago

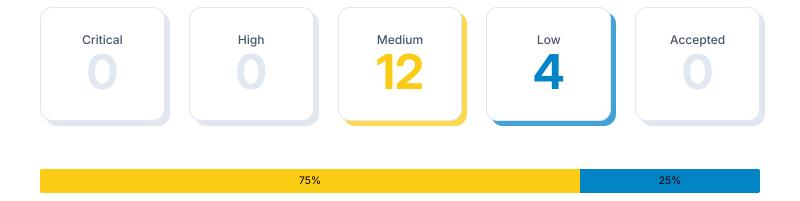
The NMAP TCP port scan discovers open TCP ports with a complete scan of ports 0 to 65535.

#### Lite Scan

Free accounts use the lite port scan which is limited to the top 100 most common ports.

## **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



### **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

Title	Severity	Open	Accepted
Open TCP Port: 21	Medium	1	0
Open TCP Port: 22	Medium	1	0
Open TCP Port: 25	Medium	1	0
Open TCP Port: 26	Medium	1	0
Open TCP Port: 53	Medium	1	0
Open TCP Port: 110	Medium	1	0
Open TCP Port: 143	Medium	1	0

Open TCP Ports Vulnerability Scan Report

Open TCP Port: 587	Medium	1	0
Open TCP Port: 993	Medium	1	0
Open TCP Port: 995	Medium	1	0
Open TCP Port: 3306	Medium	1	0
Open TCP Port: 80	Low	2	0
Open TCP Port: 443	Low	2	0

## **Vulnerability Details**

Detailed information about each potential vulnerability found by the scan.



# Open TCP Port: 21

**SEVERITY** 

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

21

#### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

22

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

25

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

26

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

53

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

110

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

143

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

465

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

587

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

993

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

995

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Medium

1 target

0 days ago

3306

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Low

2 targets

0 days ago

80

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago



SEVERITY

AFFECTED TARGETS

LAST DETECTED

PORT

Low

2 targets

0 days ago

443

### **Description**

An open port may be an expected configuration. For example, web servers use port 80 to serve websites over http and port 443 to serve websites over https. For a list of commonly used ports see https://en.wikipedia.org/wiki/List\_of\_TCP\_and\_UDP\_port\_numbers.

An unexpected open port could give unintended access to applications, data, and private networks. Open ports can also be dangerous when expected services are out of date and exploited through security vulnerabilities.

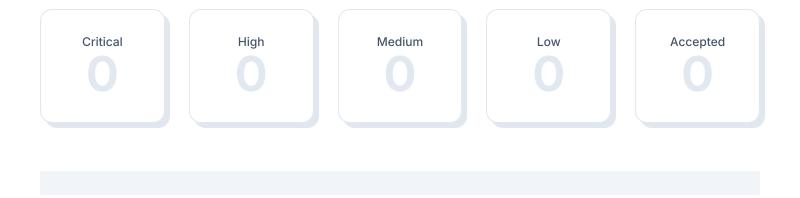
Vulnerable Target	First Detected	Last Detected
krct.ac.in	0 days ago	0 days ago
https://www.howtogeek.com/bypass-chromes-failed-virus-detected-message/	0 days ago	0 days ago

# **Open UDP Ports**

The NMAP UDP port scan discovers open ports of common UDP services

## **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



## **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

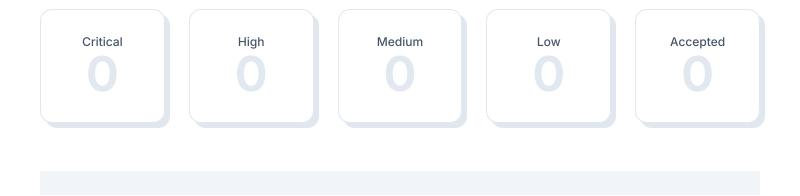
Title	Severity	Open Accepted
No vulnerabilities detected		

## Nuclei Vulnerability Scanner

Fast vulnerability scanner powered by the community-driven template engine. Detects CVEs, misconfigurations, and security issues across web applications and infrastructure.

## **Total Vulnerabilities**

Total number of vulnerabilities found by severity.



## **Vulnerabilities Breakdown**

Summary list of all detected vulnerabilities.

Title	Severity	Open Accepted
No vulnerabilities detected		

Glossary Vulnerability Scan Report

## Glossary

#### **Accepted Vulnerability**

An accepted vulnerability is one which has been manually reviewed and classified as acceptable to not fix at this time, such as a false positive scan result or an intentional part of the system's architecture.

#### **Active Web Application Vulnerabilities**

The OWASP ZAP Active Web Application scan crawls the pages of a website or web application testing for vulnerabilities and configuration weaknesses. The active scan includes all of the passive scan tests and additionally makes requests and submits forms to actively test an application for more vulnerabilities. The active scan tests for vulnerabilities such as SQL injection, remote command execution, XSS, and more.

#### **Fully Qualified Domain Name (FQDN)**

A fully qualified domain name is a complete domain name for a specific website or service on the internet. This includes not only the website or service name, but also the top-level domain name, such as .com, .org, .net, etc. For example, 'www.example.com' is an FQDN.

#### **Passive Web Application Vulnerabilities**

The OWASP ZAP Passive Web Application scan crawls the pages of a website or web application. The passive scan inspects each page as well as the requests and responses sent between the server. The passive scan checks for vulnerabilities such as cross-domain misconfigurations, insecure cookies, vulnerable Javascript dependencies, and more

#### **Network Vulnerabilities**

The OpenVAS network vulnerability scan tests servers and internet connected devices for over 150,000 vulnerabilities. OpenVAS uses the Common Vulnerability Scoring System (CVSS) to quantify the severity of findings. 0.0 is the lowest severity and 10.0 is the highest.

#### **Open TCP Ports**

The NMAP TCP port scan discovers open TCP ports with a complete scan of ports 0 to 65535.

#### **Open UDP Ports**

The NMAP UDP port scan discovers open ports of common UDP services

#### Vulnerability

A weakness in the computational logic (e.g., code) found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability. Mitigation of the vulnerabilities in this context typically involves coding changes, but could also include specification changes or even specification deprecations (e.g., removal of affected protocols or functionality in their entirety).

#### SSL/TLS Security

The SSLyze security scan tests for misconfigured SSL/TLS certificates, expired certificates, weak ciphers, and SSL/TLS vulnerabilities such as Heartbleed.

#### Target

A target represents target is a single URL, IP address, or fully qualified domain name (FQDN) that was scanned.

#### Severity

Severity represents the estimated impact potential of a particular vulnerability. Severity is divided into 5 categories: Critical, High, Medium, Low and Accepted.

#### **CVSS Score**

The CVSS 3.0 score is a global standard for evaluating vulnerabilities with a 0 to 10 scale. CVSS maps to threat levels:

0.1 - 3.9 = Low

4.0 - 6.9 = Medium

7.0 - 8.9 = High

9.0 - 10.0 = Critical

#### **EPSS Score**

The EPSS score is the estimated probability that a given vulnerability will be exploited in the wild within the next 30 days, on a 0% to 100% scale.

#### This report was prepared using

## HostedScan Security ®

### For more information, visit hostedscan.com

Founded in Seattle, Washington in 2019, HostedScan, LLC. is dedicated to making continuous vulnerability scanning and risk management much more easily accessible to more businesses.

