ZAP Scanning Report

Site: https://qa.contilink.com

Generated on Fri, 2 Sept 2022 14:16:32

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	4
Low	6
Informational	3

Alerts

Name	Risk Level	Number of Instances
Absence of Anti-CSRF Tokens	Medium	1
Content Security Policy (CSP) Header Not Set	Medium	3
Cross-Domain Misconfiguration	Medium	9
Vulnerable JS Library	Medium	1
Cookie No HttpOnly Flag	Low	2
Cookie Without Secure Flag	Low	5
Cookie with SameSite Attribute None	Low	1
Cookie without SameSite Attribute	Low	9
Cross-Domain JavaScript Source File Inclusion	Low	1
Timestamp Disclosure - Unix	Low	64
Information Disclosure - Sensitive Information in URL	Informational	2
Information Disclosure - Suspicious Comments	Informational	15
Re-examine Cache-control Directives	Informational	2

Alert Detail

Medium	Absence of Anti-CSRF Tokens
	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:

Description	* 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.
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	GET
Attack	
Evidence	<form method="post" name="login" id="login" cssClass="form-signin" commandName="${commandName}" htmlEscape="true" role="form">
Instances	1
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.
Reference	http://projects.webappsec.org/Cross-Site-Request-Forgery http://cwe.mitre.org/data/definitions/352.html
CWE Id	352
WASC Id	9

Plugin Id	10202

Medium	Content Security Policy (CSP) Header Not Set
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.
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	GET
Attack	
Evidence	
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	
URL	https://qa.contilink.com/jsp/cm8/fleet/menu/contisalesrep.jsp
Method	GET
Attack	
Evidence	
Instances	3
Solution	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header, to achieve optimal browser support: "Content-Security-Policy" for Chrome 25+, Firefox 23+ and Safari 7+, "X-Content-Security-Policy" for Firefox 4.0+ and Internet Explorer 10+, and "X-WebKit-CSP" for Chrome 14+ and Safari 6+.
Reference	https://developer.mozilla.org/en-US/docs/Web/Security/CSP/Introducing_Content_Security_Policy https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html http://www.w3.org/TR/CSP/ http://w3c.github.io/webappsec/specs/content-security-policy/csp-specification.dev.html http://www.html5rocks.com/en/tutorials/security/content-security-policy/ http://caniuse.com/#feat=contentsecuritypolicy http://content-security-policy.com/
CWE Id	693
WASC Id	15
Plugin Id	10038

Medium	Cross-Domain Misconfiguration
Description	Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovali18n
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *

URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/polyfills.js
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/runtime.js
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/styles.css
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/d9InvoiceDetails
Method	POST
Attack	
Evidence	Access-Control-Allow-Origin: *
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/updateItemD9List
Method	POST
Attack	
Evidence	Access-Control-Allow-Origin: *
Instances	9
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.
Reference	https://vulncat.fortify.com/en/detail?id=desc.config.dotnet.html5_overly_permissive_cors_policy
CWE Id	264
WASC Id	14

Plugin Id	10098

Medium	Vulnerable JS Library
Description	The identified library jquery, version 1.11.3 is vulnerable.
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	* jQuery JavaScript Library v1.11.3
Instances	1
Solution	Please upgrade to the latest version of jquery.
Reference	https://github.com/jquery/jquery/issues/2432 http://blog.jquery.com/2016/01/08/jquery-2-2-and-1-12-released/ http://research.insecurelabs.org/jquery/test/ https://blog.jquery.com/2019/04/10/jquery-3-4-0-released/ https://nvd.nist.gov/vuln/detail/CVE-2019-11358 https://nvd.nist.gov/vuln/detail/CVE-2015-9251 https://github.com/jquery/jquery/commit/753d591aea698e57d6db58c9f722cd0808619b1b https://bugs.jquery.com/ticket/11974 https://blog.jquery.com/2020/04/10/jquery-3-5-0-released/
CWE Id	829
WASC Id	
Plugin Id	10003

Low	Cookie No HttpOnly Flag
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.
URL	https://qa.contilink.com/j_spring_cas_security_check?ticket=ST-4698-dVfx0EllDLb50CCdvO75JIuQIDw-e83d9c351ead
Method	GET
Attack	
Evidence	Set-Cookie: d
URL	https://qa.contilink.com/jsp/cm8/fleet/menu/contisalesrep.jsp
Method	GET
Attack	
Evidence	Set-Cookie: locale
Instances	2
Solution	Ensure that the HttpOnly flag is set for all cookies.
Reference	https://owasp.org/www-community/HttpOnly
CWE Id	1004
WASC Id	13
Plugin Id	10010

Low	Cookie Without Secure Flag
Description	A cookie has been set without the secure flag, which means that the cookie can be accessed via unencrypted connections.
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET

Attack	
Evidence	Set-Cookie: JSESSIONID
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	Set-Cookie: SESSION
URL	https://qa.contilink.com/coc/sapinvoice/css/app.css
Method	GET
Attack	
Evidence	Set-Cookie: JSESSIONID
URL	https://qa.contilink.com/j_spring_cas_security_check?ticket=ST-4698-dVfx0EllDLb50CCdvO75JIuQIDw-e83d9c351ead
Method	GET
Attack	
Evidence	Set-Cookie: d
URL	https://qa.contilink.com/login/oauth2/code/cocgateway?code=OC-2011--VLT2O7pXrRCQrVhQ7p0vnK73YC55PXV&state=uuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%3D&nonce=g3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ
Method	GET
Attack	
Evidence	Set-Cookie: SESSION
Instances	5
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.
Reference	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
CWE Id	614
WASC Id	13
Plugin Id	10011

Low	**Cookie with SameSite Attribute None**
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.
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	POST
Attack	
Evidence	Set-Cookie: TGC
Instances	1
Solution	Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.

Reference	https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site
CWE Id	1275
WASC Id	13
Plugin Id	10054

Low	**Cookie without SameSite Attribute**
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.
URL	https://qa.contilink.com/cas/js/conti/login.js
Method	GET
Attack	
Evidence	Set-Cookie: org.springframework.web.servlet.i18n.CookieLocaleResolver.LOCALE
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	GET
Attack	
Evidence	Set-Cookie: org.springframework.web.servlet.i18n.CookieLocaleResolver.LOCALE
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fj_spring_cas_security_check
Method	GET
Attack	
Evidence	Set-Cookie: org.springframework.web.servlet.i18n.CookieLocaleResolver.LOCALE
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	Set-Cookie: JSESSIONID
URL	https://qa.contilink.com/coc/sapinvoice/css/app.css
Method	GET
Attack	
Evidence	Set-Cookie: JSESSIONID
URL	https://qa.contilink.com/j_spring_cas_security_check?ticket=ST-4698-dVfx0EllDLb50CCdvO75JluQIDw-e83d9c351ead
Method	GET
Attack	
Evidence	Set-Cookie: d
URL	https://qa.contilink.com/jsp/cm8/fleet/menu/contisalesrep.jsp
Method	GET
Attack	
Evidence	Set-Cookie: JSESSIONID
URL	https://qa.contilink.com/jsp/cm8/fleet/menu/contisalesrep.jsp

Method	GET
Attack	
Evidence	Set-Cookie: locale
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	POST
Attack	
Evidence	Set-Cookie: org.springframework.web.servlet.i18n.CookieLocaleResolver.LOCALE
Instances	9
Solution	Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.
Reference	https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site
CWE Id	1275
WASC Id	13
Plugin Id	10054

Low	**Cross-Domain JavaScript Source File Inclusion**
Description	The page includes one or more script files from a third-party domain.
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	<script async src="https://www.googletagmanager.com/gtag/js?id=UA-121373122-1"></script>
Instances	1
Solution	Ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end users of the application.
Reference	
CWE Id	829
WASC Id	15
Plugin Id	10017

Low	**Timestamp Disclosure - Unix**
Description	A timestamp was disclosed by the application/web server - Unix
URL	https://qa.contilink.com/cas/css/bootstrap/bootstrap-3.3.6.min.css
Method	GET
Attack	
Evidence	42857143
URL	https://qa.contilink.com/cas/css/bootstrap/bootstrap-3.3.6.min.css
Method	GET
Attack	
Evidence	80000000
URL	https://qa.contilink.com/cas/css/main-a4ec989fd9.css
Method	GET

Attack	
Evidence	0464027001
URL	https://qa.contilink.com/cas/css/main-a4ec989fd9.css
Method	GET
Attack	
Evidence	1866490459
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	0123456789
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	2147483647
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	62425156
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	94906265
URL	https://qa.contilink.com/cite/resources/style/main_cite-323e479a8a.css
Method	GET
Attack	
Evidence	0464027001
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	00100000
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	0123456789
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	10040064
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	

Evidence	10040166
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	10066431
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	10079232
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	10079487
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	101010256
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	12632256
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	13408767
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	13421823
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	13434828
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	13434879
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	134695760

URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16200000
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16711680
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16711935
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16737792
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16744576
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16750848
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16751052
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16763904
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16764057
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16776960
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js

Method	GET
Attack	
Evidence	16777113
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16777164
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16777215
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	16777216
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	1919054434
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	2147483647
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	268435455
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	306674912
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	318902576
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	33639248
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET

Attack	
Evidence	536870912
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	67324752
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	842412599
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	859007059
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/styles.css
Method	GET
Attack	
Evidence	00000005
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/styles.css
Method	GET
Attack	
Evidence	00000024
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/styles.css
Method	GET
Attack	
Evidence	00000042
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/styles.css
Method	GET
Attack	
Evidence	00000061
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	0007444874
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	
Evidence	07444874
URL	https://qa.contilink.com/coc/invoicepreapproval/invPreApp/invoicesPending?ref=3967702
Method	GET
Attack	

Evidence	121373122
URL	https://qa.contilink.com/jsp/cm8/fleet/menu/contisalesrep.jsp
Method	GET
Attack	
Evidence	15552000
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	0007400022
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	0007444874
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	0008266846
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	0019844639
URL	https://qa.contilink.com/coc/invoicepreapproval/api/invoicePreApprovalList/dealerD9List
Method	POST
Attack	
Evidence	87777777
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/d9InvoiceDetails
Method	POST
Attack	
Evidence	0007400022
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/d9InvoiceDetails
Method	POST
Attack	
Evidence	0007444874
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/d9InvoiceDetails
Method	POST
Attack	
Evidence	0008266846
URL	https://qa.contilink.com/coc/invoicepreapproval/api/readItemList/d9InvoiceDetails
Method	POST
Attack	
Evidence	87777777

Instances	64
Solution	Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.
Reference	http://projects.webappsec.org/w/page/13246936/Information%20Leakage
CWE Id	200
WASC Id	13
Plugin Id	10096

Informational	**Information Disclosure - Sensitive Information in URL**
Description	The request appeared to contain sensitive information leaked in the URL. This can violate PCI and most organizational compliance policies. You can configure the list of strings for this check to add or remove values specific to your environment.
URL	https://qa.contilink.com/cas/oauth2.0/callbackAuthorize?client_id=cocgateway&redirect_uri=https%3A%2F%2Fqa.contilink.com%2Flogin%2Foauth2%2Fcode%2Fcocgateway&response_type=code&state=uuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%3D&nonce=g3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ&client_name=CasOAuthClient&ticket=ST-4697-hulm-yM-vlyWvTRcdHuRqhWDF6Y-e83d9c351ead
Method	GET
Attack	
Evidence	ticket
URL	https://qa.contilink.com/j_spring_cas_security_check?ticket=ST-4698-dVfx0EllDLb50CCdvO75JIuQIDw-e83d9c351ead
Method	GET
Attack	
Evidence	ticket
Instances	2
Solution	Do not pass sensitive information in URIs.
Reference	
CWE Id	200
WASC Id	13
Plugin Id	10024

Informational	**Information Disclosure - Suspicious Comments**
Description	The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.
URL	https://qa.contilink.com/cas/js/coc_cas.js
Method	GET
Attack	
Evidence	from
URL	https://qa.contilink.com/cas/js/conti/global.js
Method	GET
Attack	
Evidence	where
URL	https://qa.contilink.com/cas/js/conti/login.js
Method	GET
Attack	

Evidence	username
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	GET
Attack	
Evidence	username
URL	https://qa.contilink.com/cas/webjars/jquery/3.6.0/jquery.min.js
Method	GET
Attack	
Evidence	username
URL	https://qa.contilink.com/cite/resources/picturefill.min.js
Method	GET
Attack	
Evidence	from
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	fixme
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	from
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	later
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	query
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	select
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js
Method	GET
Attack	
Evidence	user
URL	https://qa.contilink.com/cite/resources/script/main-cite.4fff304446e7ef00a539.js

Method	GET
Attack	
Evidence	username
URL	https://qa.contilink.com/coc/invoicepreapproval/inv-pre-app-ui/dist/cite-inv-pre-app/main.js
Method	GET
Attack	
Evidence	TODO
URL	https://qa.contilink.com/cas/login?service=https%3A%2F%2Fqa.contilink.com%2Fcas%2Foauth2.0%2FcallbackAuthorize%3Fclient_id%3Dcocgateway%26redirect_uri%3Dhttps%253A%252F%252Fqa.contilink.com%252Flogin%252Foauth2%252Fcode%252Fcocgateway%26response_type%3Dcode%26state%3DuuNyiEc-21B-mlpUoMfmToxPFzIBNLwOzN87z17hip8%253D%26nonce%3Dg3DhQzLXf7p8xJThkABZ-Yil17TFKR-n4oL_bP0hnTQ%26client_name%3DCasOAuthClient
Method	GET
Attack	
Evidence	from
Instances	15
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Reference	
CWE Id	200
WASC Id	13
Plugin Id	10027

Informational	**Re-examine Cache-control Directives**
Description	The cache-control header has not been set properly or is missing, allowing the browser and proxies to cache content. For static assets like css, js, or image files this might be intended, however, the resources should be reviewed to ensure that no sensitive content will be cached.
URL	https://qa.contilink.com/cite/servlet/messages/924064?_=
Method	GET
Attack	
Evidence	max-age=86400
URL	https://qa.contilink.com/cite/servlet/settings/924064?_=
Method	GET
Attack	
Evidence	max-age=86400
Instances	2
Solution	For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".
Reference	https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html#web-content-caching https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cache-Control
CWE Id	525
WASC Id	13
Plugin Id	10015