curl 60 ssl certificate problem self signed certificate

curl 60 ssl certificate problem self signed certificate is a common error encountered when using the curl command-line tool to make HTTPS requests to servers with self-signed SSL certificates. This error arises because curl performs strict SSL certificate verification by default, and self-signed certificates are not trusted by default certificate authorities. Understanding the causes of this issue, how SSL certificates work, and the methods to resolve or bypass the error is essential for developers, system administrators, and users working with secure connections. This article delves into the technical background of the curl 60 SSL certificate problem, explores its implications, and provides comprehensive solutions to address it effectively. Additionally, best practices for managing self-signed certificates and maintaining secure communications are discussed to ensure a balance between security and functionality.

Understanding the curl 60 SSL Certificate Problem

The curl 60 SSL certificate problem self signed certificate error indicates that curl has detected an issue with the SSL certificate presented by the server during the HTTPS handshake. Specifically, curl expects the server's SSL certificate to be signed by a trusted Certificate Authority (CA). When the server uses a self-signed certificate, curl cannot verify its authenticity against a recognized CA, resulting in error code 60. This behavior protects users from potential man-in-the-middle attacks by ensuring certificates are trusted and valid.

What Does Error 60 Mean?

Error 60 in curl corresponds to the CURLE_SSL_CACERT error, which signals that the certificate verification failed because the certificate is not trusted or cannot be verified. This error commonly surfaces when connecting to servers with self-signed certificates or during development and testing phases where proper CA-signed certificates are unavailable. Understanding this error is the first step in diagnosing and resolving the issue.

Typical Scenarios Causing the Error

The curl 60 SSL certificate problem self signed certificate frequently appears in various scenarios, including:

- Connecting to internal or development servers with self-signed certificates.
- Accessing APIs or services that use custom or corporate-issued certificates not in the default trusted store.
- Testing SSL configurations before obtaining certificates from trusted CAs.

Causes of the Self-Signed Certificate Error

Several factors contribute to the curl 60 SSL certificate problem self signed certificate error. These causes revolve around SSL/TLS certificate verification mechanisms and trust chains as implemented by curl and underlying SSL libraries like OpenSSL.

Self-Signed Certificates Are Not Trusted by Default

Self-signed certificates are generated and signed by the entity owning the server rather than a recognized third-party CA. Because they lack an external trust anchor, operating systems and tools like curl do not trust them automatically. This lack of trust triggers verification failures during SSL handshakes.

Missing or Outdated CA Certificate Bundle

curl relies on a bundle of trusted CA certificates to verify server certificates. If this bundle is missing, outdated, or improperly configured, even valid certificates may trigger errors. This situation can also cause the curl 60 SSL certificate problem self signed certificate to appear incorrectly.

Incorrect System Date and Time

SSL certificates have validity periods. If the client system's date and time are incorrect, certificates may appear expired or not yet valid, causing curl's verification process to fail and generate error 60.

How SSL Certificates and Verification Work

Understanding the architecture behind SSL certificates and their verification process helps clarify why the curl 60 SSL certificate problem self signed certificate occurs and how it can be addressed.

Role of Certificate Authorities (CAs)

Certificate Authorities are trusted organizations that issue SSL certificates after validating the identity of the requester. These certificates create a trust chain from the server's certificate up to a root CA recognized by client systems. This chain ensures that the server is authentic and mitigates risks of impersonation.

Certificate Chain and Trust Stores

When curl connects to an HTTPS server, it receives the server's certificate along with any intermediate certificates. curl uses its trusted CA bundle to verify this chain, ensuring the certificate was issued by a trusted CA. If any link in the chain is invalid or missing, verification fails, causing error 60.

Self-Signed Certificates and Trust

Self-signed certificates are not part of any trusted CA chain. They are signed by the server itself, so no external authority vouches for their authenticity. This lack of external validation is why curl and other clients do not trust them by default.

Methods to Resolve curl 60 SSL Certificate Problem

Several approaches exist to fix or work around the curl 60 SSL certificate problem self signed certificate error. The appropriate method depends on the security requirements and context of the connection.

Adding the Self-Signed Certificate to the Trusted Store

One secure solution is to add the server's self-signed certificate to the client's trusted CA bundle. This action informs curl to trust the certificate during verification.

1. Obtain the self-signed certificate in PEM format from the server.

2. Add the certificate to the local CA bundle used by curl or the operating system.

3. Configure curl to use the updated CA bundle if necessary.

Using curl Options to Bypass Verification

For testing or development, curl provides options to bypass SSL certificate verification:

- `--insecure` or `-k`: This option tells curl to skip certificate verification entirely, suppressing error 60.

- `--cacert [file]`: Specifies a custom CA bundle file containing the self-signed certificate.

While these methods temporarily solve the problem, they reduce the security of the connection and should not be used in production environments.

Updating curl and OpenSSL

Ensuring that curl and its SSL backend (such as OpenSSL) are up-to-date can resolve the issue if caused by outdated or incompatible libraries. Updated versions include improved certificate handling and trust store management.

Checking System Date and Time

Verifying and correcting the system clock can prevent erroneous certificate validity issues. Synchronizing time with a reliable source like NTP is recommended.

Security Implications of Using Self-Signed Certificates

Using self-signed certificates carries inherent security risks that must be carefully considered before deployment. These implications explain why the curl 60 SSL certificate problem self signed certificate error exists as a safeguard.

Risk of Man-in-the-Middle Attacks

Without third-party validation, self-signed certificates can be easily spoofed. Attackers can intercept and manipulate data if clients blindly trust such certificates. This risk underscores the importance of proper certificate management.

Limited Trust and Compatibility

Self-signed certificates are not automatically trusted by browsers, operating systems, or tools like curl. This limitation can lead to connectivity issues and user warnings, affecting usability and trustworthiness.

Appropriate Use Cases

Despite risks, self-signed certificates are useful in controlled environments such as internal networks, development, and testing scenarios where security risks are managed, and trust can be established manually.

Best Practices for Managing Self-Signed SSL Certificates

Proper management of self-signed certificates can mitigate many issues related to the curl 60 SSL certificate problem self signed certificate and enhance overall security posture.

Use Strong Cryptographic Parameters

Generate self-signed certificates using strong encryption algorithms and adequate key lengths to ensure robust security.

Distribute Certificates Securely

Ensure that self-signed certificates are distributed and installed securely on client systems to establish trust without exposing them to interception or tampering.

Maintain Certificate Validity

Set reasonable expiration dates and renew certificates promptly to avoid unexpected verification failures.

Document and Automate Certificate Handling

Maintain clear documentation on certificate usage and automate certificate deployment and updates where possible to reduce human error.

Consider Using Private CA Solutions

For larger environments, deploying an internal CA to issue and manage certificates provides a scalable and secure alternative to self-signed certificates.

Frequently Asked Questions

What does the 'curl 60 SSL certificate problem: self signed certificate' error mean?

This error means that curl does not trust the SSL certificate presented by the server because it is self-signed and not verified by a recognized Certificate Authority (CA).

How can I bypass the 'curl 60 SSL certificate problem: self signed certificate' error?

You can bypass this error by using the curl option '-k' or '--insecure', which tells curl to ignore certificate validation errors. For example: curl -k https://example.com

Is it safe to ignore the 'self signed certificate' error in curl?

Ignoring this error can expose you to security risks like man-in-the-middle attacks. It is safer to use a valid SSL certificate or add the self-signed certificate to your trusted certificates store if you control the server.

How do I add a self-signed certificate to curl's trusted certificates?

You can add the self-signed certificate to a local certificate file and use the '--cacert' option in curl to specify that file, e.g., curl --cacert /path/to/self-signed.crt https://example.com

Can I fix the 'curl 60 SSL certificate problem' by updating curl or CA certificates?

Yes, updating curl and the CA certificates bundle on your system can help, but if the certificate is truly self-signed and not added to the trusted store, the error will persist.

How do I check if a server uses a self-signed certificate causing curl

error 60?

You can use the command 'openssl s_client -connect hostname:443' and inspect the certificate details. If the issuer and subject are the same, it is likely self-signed.

Why does curl reject self-signed certificates by default?

Curl rejects self-signed certificates by default to prevent security risks, ensuring that the connection is secure and the server identity is verified by a trusted CA.

Can I configure curl globally to trust self-signed certificates?

You can configure curl to trust specific self-signed certificates by adding them to the system's trusted CA store or by setting the CURL_CA_BUNDLE environment variable to point to a custom CA bundle including the self-signed cert.

What is the difference between self-signed certificates and certificates from a CA?

Self-signed certificates are created and signed by the entity using them, without third-party validation. Certificates from a CA are signed by a trusted authority, providing verified identity and trust.

How do I generate a self-signed certificate for testing purposes with curl?

You can generate a self-signed certificate using OpenSSL with commands like: openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nodes. This certificate can be used in test servers but will trigger curl error 60 unless trusted explicitly.

Additional Resources

1. *Mastering cURL and SSL: Troubleshooting Self-Signed Certificate Issues*
This book provides an in-depth exploration of cURL, focusing on SSL and TLS protocols. It guides readers through common problems, including the notorious self-signed certificate error, offering practical solutions and workarounds. With real-world examples, users learn how to configure cURL for secure communication and bypass certificate validation when necessary.

2. *Practical SSL/TLS for Developers: Handling Self-Signed Certificates with cURL*
Aimed at developers, this book demystifies SSL/TLS concepts and explains how self-signed certificates impact secure HTTP requests. It includes step-by-step instructions on using cURL to interact with servers presenting self-signed certificates, along with best practices for maintaining security without compromising functionality.

3. *cURL Essentials: Secure Transfers and Certificate Management*
Covering the essentials of cURL commands and SSL certificate management, this guide helps readers understand the intricacies of secure data transfers. It addresses common SSL certificate problems, including self-signed certificate warnings, and shows how to properly configure trust stores and certificate authorities.

4. *Understanding SSL Certificate Errors in Command-Line Tools*
This book focuses on SSL certificate errors encountered in command-line tools like cURL, OpenSSL, and wget. It explains the causes behind errors such as "self-signed certificate" and provides troubleshooting techniques to resolve these issues without compromising security.

5. *Networking Security with cURL: Overcoming SSL Challenges*
Designed for network administrators and security professionals, this book delves into SSL challenges encountered during data transfers with cURL. It emphasizes dealing with self-signed certificates, certificate pinning, and secure client-server authentication, ensuring reliable and secure network communications.

6. *Hands-On Guide to SSL Certificates and cURL Integration*

This hands-on guide equips readers with practical knowledge to integrate SSL certificates into their cURL requests. It includes tutorials on generating self-signed certificates, configuring cURL options to trust these certificates, and securing API communications in development and testing environments.

7. *Debugging SSL Issues in Web Development: cURL and Beyond*
Targeted at web developers, this book explores SSL debugging techniques using cURL and other tools. It covers common SSL pitfalls such as self-signed certificate errors, expired certificates, and mismatched hostnames, providing effective strategies to diagnose and resolve these problems quickly.

8. *Secure API Consumption with cURL: Managing Certificates and Errors*
This resource focuses on securely consuming APIs using cURL while handling SSL certificate challenges. It explains how to manage self-signed certificates, configure cURL options to bypass or verify certificates, and maintain security when working with private or development APIs.

9. *The Developer's Handbook to SSL Certificates and cURL*
This handbook serves as a comprehensive reference for developers working with SSL certificates and cURL. It covers certificate creation, validation, common errors like self-signed certificate warnings, and practical tips to ensure smooth and secure command-line HTTP(S) operations.

Curl 60 Ssl Certificate Problem Self Signed Certificate

curl 60 ssl certificate problem self signed certificate: Practical Go Amit Saha, 2021-09-11 YOUR PRACTICAL, HANDS-ON GUIDE TO WRITING APPLICATIONS USING GO Google announced the Go programming language to the public in 2009, with the version 1.0 release announced in 2012. Since its announcement to the community, and the compatibility promise of the 1.0 release, the Go language has been used to write scalable and high-impact software programs ranging from command-line applications and critical infrastructure tools to large-scale distributed systems. It's speed, simplicity, and reliability make it a perfect choice for developers working in various domains. In Practical Go - Building Scalable Network + Non-Network Applications, you will learn to use the Go programming language to build robust, production-ready software applications. You will learn

just enough to building command line tools and applications communicating over HTTP and gRPC. This practical guide will cover: Writing command line applications Writing a HTTP services and clients Writing RPC services and clients using gRPC Writing middleware for network clients and servers Storing data in cloud object stores and SQL databases Testing your applications using idiomatic techniques Adding observability to your applications Managing configuration data from your applications You will learn to implement best practices using hands-on examples written with modern practices in mind. With its focus on using the standard library packages as far as possible, Practical Go will give you a solid foundation for developing large applications using Go leveraging the best of the language's ecosystem.

curl 60 ssl certificate problem self signed certificate: *Cloud Native Go* Matthew A. Titmus, 2024-10-14 Learn how to use Go's strengths to develop services that are scalable and resilient even in an unpredictable environment. With this book's expanded second edition, Go developers will explore the composition and construction of cloud native applications, from lower-level Go features and mid-level patterns to high-level architectural considerations. Each chapter in this new edition builds on the lessons of the previous chapter, taking intermediate to advanced developers through Go to construct a simple but fully featured distributed key-value store. You'll learn about Go generics, dependability and reliability, memory leaks, and message-oriented middleware. New chapters on security and distributed state delve into critical aspects of developing secure distributed cloud native applications. With this book you will: Learn the features that make Go an ideal language for building cloud native software Understand how Go solves the challenges of designing scalable distributed services Design and implement a reliable cloud native service by leveraging Go's lower-level features such as channels and goroutines Apply patterns, abstractions, and tooling to effectively build and manage complex distributed systems Overcome stumbling blocks when using Go to build and manage a cloud native service

curl 60 ssl certificate problem self signed certificate: Web Security Testing Cookbook Paco Hope, Ben Walther, 2008-10-14 Offering developers an inexpensive way to include testing as part of the development cycle, this cookbook features scores of recipes for testing Web applications, from relatively simple solutions to complex ones that combine several solutions.

curl 60 ssl certificate problem self signed certificate: *Containers for Developers Handbook* Francisco Javier Ramírez Urea, 2023-11-28 Effortlessly create and manage complex multi-component applications based on Docker containers Key Features Gain a clear understanding of software containers from the SecDevOps perspective Master the construction of application pieces within containers to achieve a seamless life cycle Prepare your applications to run smoothly and with ease in complex container orchestrators Purchase of the print or Kindle book includes a free PDF eBook Book DescriptionDevelopers are changing their deployment artifacts from application binaries to container images, giving rise to the need to build container-based apps as part of their new development workflow. Managing an app's life cycle is complex and requires effort—this book will show you how to efficiently develop, share, and execute applications. You'll learn how to automate the build and delivery process using CI/CD tools with containers as container orchestrators manage the complexity of running cluster-wide applications, creating infrastructure abstraction layers, while your applications run with high availability, resilience, and persistence. As you advance, you'll develop, test, and debug applications on your desktop and get them ready to run in production with optimal security standards, using deployment patterns and monitoring tools to help identify common issues. You'll also review deployment patterns that'll enable you to solve common deployment problems, providing high availability, scalability, and security to your applications. Finally, you'll explore different solutions to monitor, log, and instrument your applications as per open-source community standards. By the end of this book, you'll be able to manage your app's life cycle by implementing CI/CD workflows using containers to automate the building and delivery of its components.What you will learn Find out how to build microservices-based applications using containers Deploy your processes within containers using Docker features Orchestrate multi-component applications on standalone servers Deploy

applications cluster-wide in container orchestrators Solve common deployment problems such as persistency or app exposure using best practices Review your application’s health and debug it using open-source tools Discover how to orchestrate CI/CD workflows using containers Who this book is forThis book is for developers and DevOps engineers looking to learn about the implementation of containers in application development, especially DevOps engineers who deploy, monitor, and maintain container-based applications running on orchestrated platforms. In general, this book is for IT professionals who want to understand Docker container-based applications and their deployment. A basic understanding of coding and frontend-backend architectures is needed to follow the examples presented in this book.

Related to curl 60 ssl certificate problem self signed certificate

What is the meaning of "curl -k -i -X" in Linux? When you use curl to access a web page it is actually sending the GET request to the server. There are other kinds of request that can be used and -X is the way to specify this.

bash - Curl bad URL (3) - Unix & Linux Stack Exchange Both the above scripts concatenates all files given as arguments on the command line, and passes the output to curl, one line at a time. Note that I have also corrected the HTTP

How to fix curl sslv3 alert handshake failure? - Unix & Linux Stack How do I ignore or force the certificate using curl command line? When using wget seems to work fine. Also works when testing with openssl as below: $ openssl s_client -connect

How to send multiline data in curl body within bash script? I am trying to send multi-line comment in the curl body from bash script. Below is my curl invocation. #!/bin/bash temp="This is sample data: 2019/05/21 03:33:04 This is 2nd

How to trust self-signed certificate in cURL command line? 1 If you save off the self-signed.crt from your server, you can pass it to curl via "--cacert self-signed.crt" and curl will validate the certificate of your server using the given CA Cert

502 Bad Gateway when curl is talking to API A curl command in a Bash script (called by cron) asks a web service/API, on another device/server (hosts both the API and the SQL database), to perform an operation on

curl - Adding a self-signed certificate to the "trusted list" - Unix I've generated a self-signed certificate for my build server and I'd like to globally trust the certificate on my machine, as I created the key myself and I'm sick of seeing warnings. I'm on

How to use curl -w option to redirect the output to a different file How to use curl -w option to redirect the output to a different file descriptor from stdout to avoid appending it at the end of curl response? Ask Question Asked 3 years ago

Why my curl gets stuck at getting anything from some domains? The curl command inside WSL2 hangs for some domains (like youtube.com) and it runs well for other domains (like google.com). It turns out the reason is the MTU size gap

CURL request using .netrc file - Unix & Linux Stack Exchange 15 As I understand the man page (of curl), the option -n just enables looking for a .netrc file, but it does not expect the file path of this file. This is the option --netrc-file. From the

Domino's Pizza - La pizza como tu querías Pide tu Domino's Pizza favorita a domicilio o a recoger en tu tienda más cercana. Disfruta de una experiencia única con nuestras pizzas de auténtica masa e ingredientes frescos

Pizzerías Carlos: La pizza que recordabas Elige la pizza a tu gusto, escoge la masa, tamaño e ingredientes y te la hacemos a medida. Pide tu pizza favorita a domicilio o a recoger en tu tienda más cercana. Disfruta de la mejor pizza

Las mejores pizzas cerca de mí - TheFork Descubre la mejor pizza cerca de ti con TheFork. Consulta las opiniones de restaurantes de nuestra comunidad y haz tu reserva online ya

Papa Johns | La mejor pizza con los mejores ingredientes Disfruta de la mejor pizza a domicilio o a recoger en tu pizzería Papa Johns más cercana. Masa fresca e ingredientes naturales. Descubre todas las ofertas

Pizza a domicilio cerca de mí en Madrid - Pide con Uber Eats ¿Qué lugares que ofrecen entregas de Pizza a domicilio están abiertos ahora cerca de mí en Madrid? En esta página te mostraremos en todo momento los lugares de Pizza que estén

10 pizzerías de Madrid donde comer auténtica pizza italiana - ELLE Nosotros los reunimos para ti. Hemos seleccionado los diez mejores sitios para comer pizza en Madrid. Son tan deliciosas que incluso llegan a competir con las mejores de

Pizza a domicilio cerca de mí en Madrid | Pizzerias en Just Eat ¡Descubre las mejores pizzerías a domicilio cerca de ti en Madrid! Puedes pedir una pizza a domicilio en Madrid ahora mismo, y así disfrutarás de un delicioso y nutritivo plato aderezado

Pizza y comida a domicilio | Pedidos online - Telepizza Pide online tu pizza favorita y disfruta de nuestra deliciosa comida a domicilio. Consulta todas nuestras ofertas y promociones

Pizzas a domicilio y para llevar | Pizza Hut Pide tu pizza favorita de Pizza Hut a domicilio o a recoger en tu tienda más cercana. Disfruta de nuestras pizzas y complétalo con entrantes, postres y bebidas

Localiza tu tienda Allô Pizza Buscas tu pizzería para hacer tu pedido online. ¡Entra y encuentra tus Allô Pizza más cercana!

What is the meaning of "curl -k -i -X" in Linux? When you use curl to access a web page it is actually sending the GET request to the server. There are other kinds of request that can be used and -X is the way to specify this.

bash - Curl bad URL (3) - Unix & Linux Stack Exchange Both the above scripts concatenates all files given as arguments on the command line, and passes the output to curl, one line at a time. Note that I have also corrected the HTTP

How to fix curl sslv3 alert handshake failure? - Unix & Linux Stack How do I ignore or force the certificate using curl command line? When using wget seems to work fine. Also works when testing with openssl as below: $ openssl s_client -connect

How to send multiline data in curl body within bash script? I am trying to send multi-line comment in the curl body from bash script. Below is my curl invocation. #!/bin/bash temp="This is sample data: 2019/05/21 03:33:04 This is 2nd

How to trust self-signed certificate in cURL command line? 1 If you save off the self-signed.crt from your server, you can pass it to curl via "--cacert self-signed.crt" and curl will validate the certificate of your server using the given CA Cert

502 Bad Gateway when curl is talking to API A curl command in a Bash script (called by cron) asks a web service/API, on another device/server (hosts both the API and the SQL database), to perform an operation on

curl - Adding a self-signed certificate to the "trusted list" - Unix I've generated a self-signed certificate for my build server and I'd like to globally trust the certificate on my machine, as I created the key myself and I'm sick of seeing warnings. I'm on

How to use curl -w option to redirect the output to a different file How to use curl -w option to redirect the output to a different file descriptor from stdout to avoid appending it at the end of curl response? Ask Question Asked 3 years ago

Why my curl gets stuck at getting anything from some domains? The curl command inside WSL2 hangs for some domains (like youtube.com) and it runs well for other domains (like google.com). It turns out the reason is the MTU size gap

CURL request using .netrc file - Unix & Linux Stack Exchange 15 As I understand the man page (of curl), the option -n just enables looking for a .netrc file, but it does not expect the file path of this file. This is the option --netrc-file. From the

What is the meaning of "curl -k -i -X" in Linux? When you use curl to access a web page it is actually sending the GET request to the server. There are other kinds of request that can be used

and -X is the way to specify this.

bash - Curl bad URL (3) - Unix & Linux Stack Exchange Both the above scripts concatenates all files given as arguments on the command line, and passes the output to curl, one line at a time. Note that I have also corrected the HTTP

How to fix curl sslv3 alert handshake failure? - Unix & Linux Stack How do I ignore or force the certificate using curl command line? When using wget seems to work fine. Also works when testing with openssl as below: $ openssl s_client -connect

How to send multiline data in curl body within bash script? I am trying to send multi-line comment in the curl body from bash script. Below is my curl invocation. #!/bin/bash temp="This is sample data: 2019/05/21 03:33:04 This is 2nd

How to trust self-signed certificate in cURL command line? 1 If you save off the self-signed.crt from your server, you can pass it to curl via "--cacert self-signed.crt" and curl will validate the certificate of your server using the given CA Cert

502 Bad Gateway when curl is talking to API A curl command in a Bash script (called by cron) asks a web service/API, on another device/server (hosts both the API and the SQL database), to perform an operation on

curl - Adding a self-signed certificate to the "trusted list" - Unix I've generated a self-signed certificate for my build server and I'd like to globally trust the certificate on my machine, as I created the key myself and I'm sick of seeing warnings. I'm on

How to use curl -w option to redirect the output to a different file How to use curl -w option to redirect the output to a different file descriptor from stdout to avoid appending it at the end of curl response? Ask Question Asked 3 years ago

Why my curl gets stuck at getting anything from some domains? The curl command inside WSL2 hangs for some domains (like youtube.com) and it runs well for other domains (like google.com). It turns out the reason is the MTU size gap

CURL request using .netrc file - Unix & Linux Stack Exchange 15 As I understand the man page (of curl), the option -n just enables looking for a .netrc file, but it does not expect the file path of this file. This is the option --netrc-file. From the

What is the meaning of "curl -k -i -X" in Linux? When you use curl to access a web page it is actually sending the GET request to the server. There are other kinds of request that can be used and -X is the way to specify this.

bash - Curl bad URL (3) - Unix & Linux Stack Exchange Both the above scripts concatenates all files given as arguments on the command line, and passes the output to curl, one line at a time. Note that I have also corrected the HTTP

How to fix curl sslv3 alert handshake failure? - Unix & Linux Stack How do I ignore or force the certificate using curl command line? When using wget seems to work fine. Also works when testing with openssl as below: $ openssl s_client -connect

How to send multiline data in curl body within bash script? I am trying to send multi-line comment in the curl body from bash script. Below is my curl invocation. #!/bin/bash temp="This is sample data: 2019/05/21 03:33:04 This is 2nd

How to trust self-signed certificate in cURL command line? 1 If you save off the self-signed.crt from your server, you can pass it to curl via "--cacert self-signed.crt" and curl will validate the certificate of your server using the given CA Cert

502 Bad Gateway when curl is talking to API A curl command in a Bash script (called by cron) asks a web service/API, on another device/server (hosts both the API and the SQL database), to perform an operation on

curl - Adding a self-signed certificate to the "trusted list" - Unix I've generated a self-signed certificate for my build server and I'd like to globally trust the certificate on my machine, as I created the key myself and I'm sick of seeing warnings. I'm on

How to use curl -w option to redirect the output to a different file How to use curl -w option to redirect the output to a different file descriptor from stdout to avoid appending it at the end of

curl response? Ask Question Asked 3 years ago
Why my curl gets stuck at getting anything from some domains? The curl command inside WSL2 hangs for some domains (like youtube.com) and it runs well for other domains (like google.com). It turns out the reason is the MTU size gap
CURL request using .netrc file - Unix & Linux Stack Exchange 15 As I understand the man page (of curl), the option -n just enables looking for a .netrc file, but it does not expect the file path of this file. This is the option --netrc-file. From the

Related to curl 60 ssl certificate problem self signed certificate

CloudPanel installations use the same SSL certificate private key (Bleeping Computer2y) Self-hosted web administration solution CloudPanel was found to have several security issues, including using the same SSL certificate private key across all installations and unintentional
CloudPanel installations use the same SSL certificate private key (Bleeping Computer2y) Self-hosted web administration solution CloudPanel was found to have several security issues, including using the same SSL certificate private key across all installations and unintentional

Back to Home: http://www.devensbusiness.com