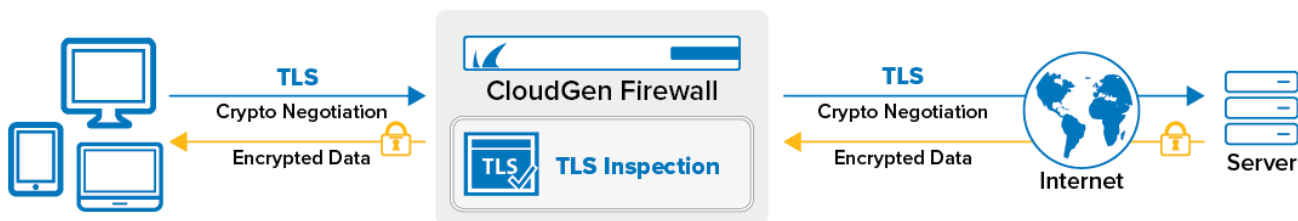


TLS Inspection in the Firewall

<https://campus.barracuda.com/doc/98210321/>

TLS Inspection decrypts TLS connections so the firewall can allow Application Control features, such as the Virus Scanner and ATP, to scan traffic that would otherwise not be visible to the firewall service. Using TLS Inspection allows the admin to enforce TLS security at the firewall by blocking outdated ciphers or refusing connections attempting to use outdated TLS versions. For outbound SSL Inspection, the firewall can also handle TLS validation errors, depending on the TLS error policy assigned to the matching access rule of the TLS session. TLS Inspection is supported for Pass, Map, and Dst NAT access rules. Not supported are TLS connections that require client certificate authentication.

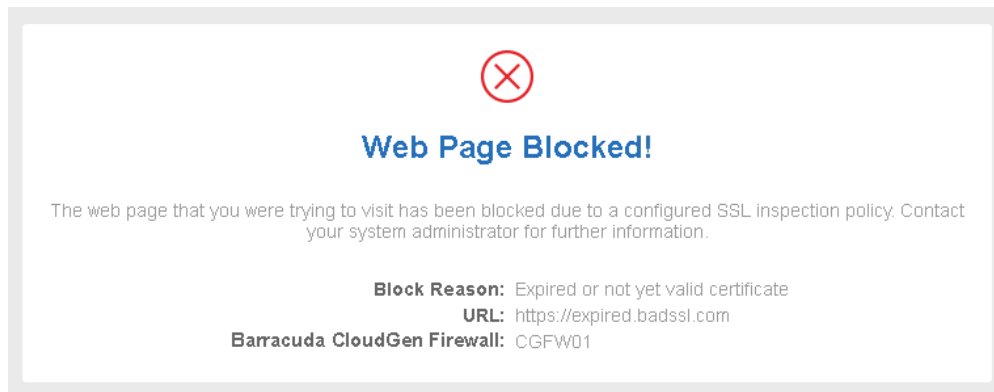


Enforce Ciphers and Minimum TLS Version

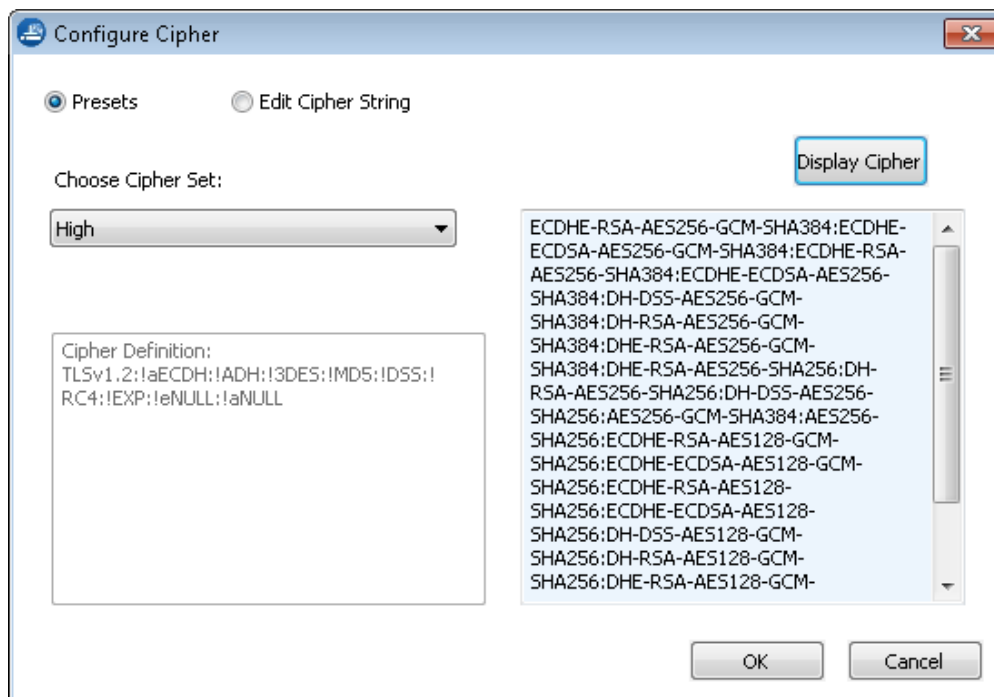
The ciphers and the TLS version for the TLS connection are negotiated between the client and the firewall and the firewall and the server. The negotiated settings for the TLS connection between the firewall and the client and the connection between the firewall and the server always strive to use the most secure cipher and TLS version possible, which can result in different encryption settings for each connection. For example, a TLS connection from a client that only supports TLSv3, while the server on the other side of the firewall supports TLS 1.2, will result in an TLSv3 connection to the firewall and a TLS 1.2 connection from the firewall to the server, if the settings of the TLS Inspection policy allow these connections.

TLS Inspection on the firewall allows the admin to define the minimum supported TLS version and a suite of allowed ciphers on a per-access-rule basis. By using different TLS Inspection policy objects, traffic for legacy applications without support for the newest TLS version can continue to be used without having to reduce the more stringent security policy used for all other TLS connections. Three preset cipher sets (HIGH, MEDIUM, and LOW) are available. The ciphers included in the presets are defined by the TLS version on the firewall. If needed, the admin can extend the cipher sets to keep up-to-date with the latest security vulnerabilities.

Connections that do not meet the requirements set in the TLS Inspection policy object are blocked by the firewall. If it is an outgoing HTTPS connection, the client is redirected to a block page.



It is possible to modify the default TLS cipher string (LOW, MEDIUM, and HIGH) by appending your own TLS-compatible string. Use the syntax compatible with the TLS version used by the firewall. The syntax of the cipher string may have to be changed if the underlying TLS version has changed during the update. The TLS version is listed in the Firewall/SSL log when log verbosity is set to **info** in the advanced TLS Inspection setting in the Security Policy settings.



Certificate Revocation Checks

Since clients behind a firewall using TLS Inspection cannot complete their own revocation check, the firewall handles the revocation check during the TLS Inspection process. To ensure that certificates are not treated as valid that have been revoked before their expiration date, the firewall queries the revocation status of the certificate in the following order:

- OCSP Stapling
- OCSP
- CRL

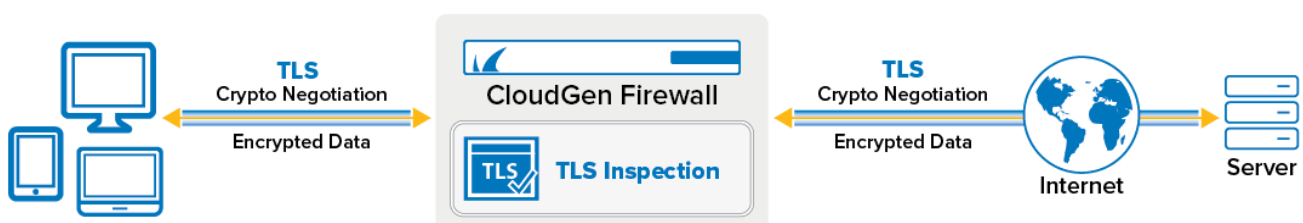
OCSP stapling is an offline check; for OCSP and CRL, the firewall must be online. The first revocation check that returns a valid result is used. To reduce the load on the OCSP or CRL servers, the revocation check is done either once per 24h period, or immediately after a TLS Inspection configuration change. If an error occurs during the revocation check, it is repeated after the configured **Revocation Fail Retry Interval (Assigned Services > Firewall > Security Policy > Enable TLS Inspection > Advanced)**. Depending on the fail close or fail open policies, TLS connections are either blocked or allowed when the revocation check fails due to operational reasons, such as an unreachable OCSP server.

Exceptions from TLS Inspection

Traffic falling into one of the following three categories is exempt from TLS Inspection:

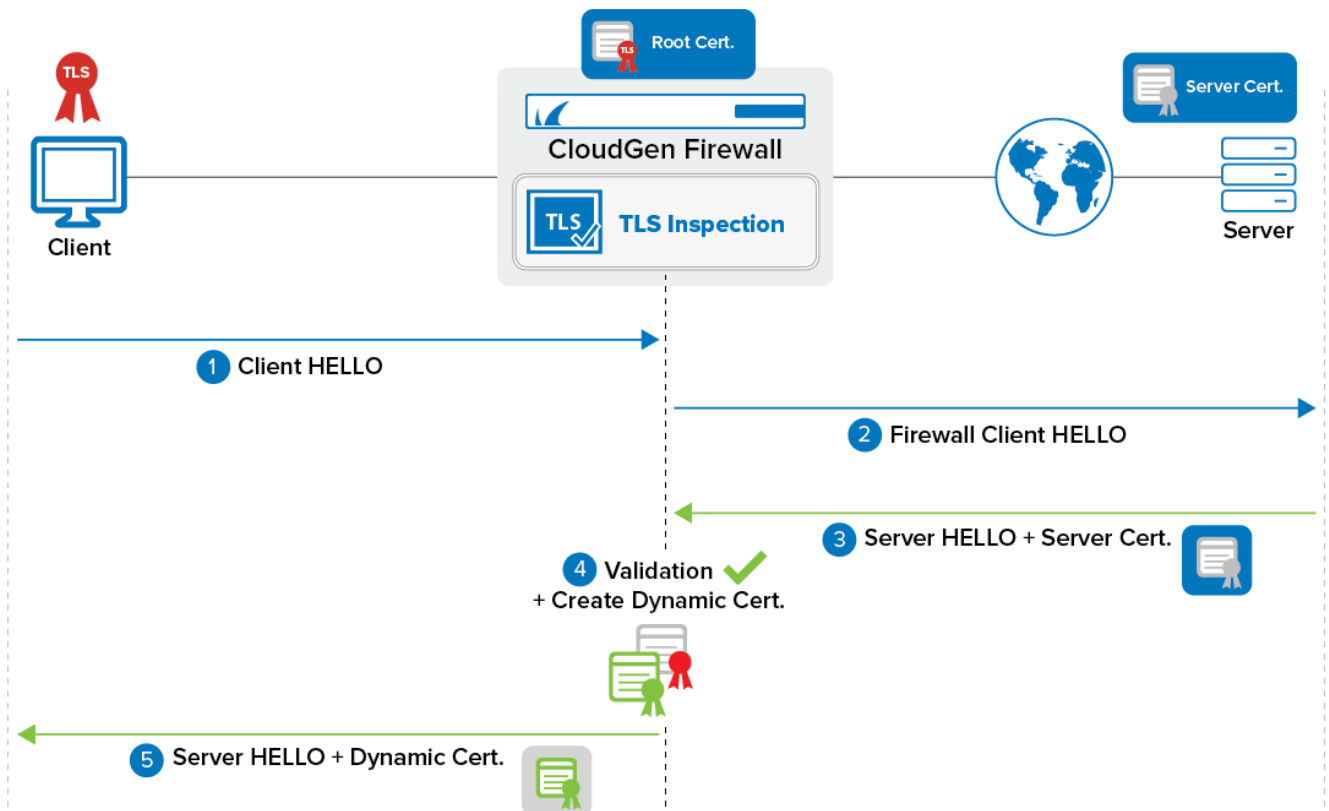
- (outbound TLS Inspection only) Applications marked as **not interceptable** in the application object properties. Go to the application object on the **Forwarding Rules > Applications** page to check if an application is interceptable.
- Domains allow-listed in the Security Policy settings.
- URL categories allow-listed in the Security Policy settings.

Outbound TLS Inspection



Outbound TLS Interception inspects TLS connections from clients behind the firewall to the Internet. The firewall decrypts the TLS traffic to allow Application Control features such as the URL Filter, Virus Scanner, or File Content policy to scan the traffic. The firewall dynamically creates a certificate and signs it with the TLS Inspection root certificate. The traffic is re-encrypted and forwarded to the destination. For the client to accept the intercepted TLS session with the dynamic self-signed certificate, the TLS Inspection root certificate must be placed in the Trusted Root Certificates store on every client behind the firewall.

Outbound TLS Inspection Process



1. The client sends client HELLO to server. This includes supported TLS versions and ciphers.
2. Connection is intercepted (blocked) by the firewall. Firewall sends client HELLO to the server using the ciphers and supported TLS versions in the TLS Inspection policy.
3. Server replies with the server HELLO, which includes the server's certificate.
4. The firewall validates the certificate. If invalid, the connection is terminated.
5. The firewall generates a dynamic certificate. Wherever possible, the dynamic certificate is created with the same values as the server certificate. The dynamic certificate is signed by the root certificate configured in the Security Policy settings.
6. Server HELLO is sent from the firewall to the client using the dynamic certificate.
7. The client trusts the root certificate of the firewall and the connection is established.

TLS Policy Validation Errors

By default, TLS errors that occur during the TLS handshake cause the firewall to terminate the connection. Some validation errors can be handled based on the policy set in the TLS Inspection policy object.

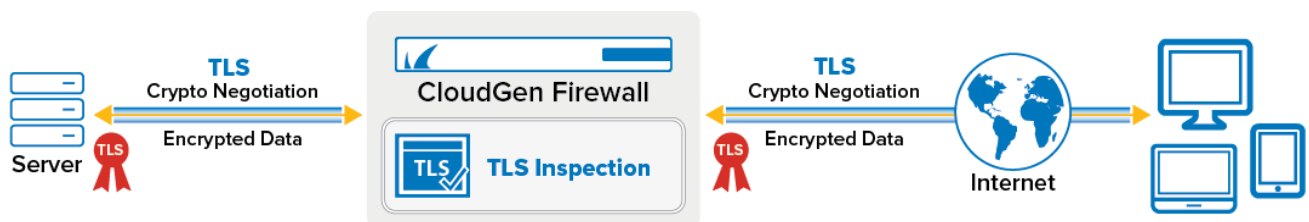
- **Pass Error to Client (default)** – A purposefully invalid TLS certificate is generated for the client, causing an error message to the client.
- **Hide Error from Client** – The client receives a valid TLS certificate, even if the TLS connection causes a TLS error on the firewall.
- **Block** – The connection is blocked on the firewall. HTTPS connections are redirected to a block page.

The following TLS validation errors are configurable:

- **Self-Signed Certificates** – Certificates with an untrusted root CA are considered self-signed.
- **Untrusted Certificates** – Certificates with one or more untrusted or missing certificates in the trust chain.
- **Expired or Not Yet Valid Certificates** – Certificates that are outside of the validity period as defined by the start and expiration date.
- **Revoked Certificates** – Certificates for which the revocation status check returns a hold or revoked status.
- **Corrupted Certificates** – Certificates that fail the integrity check indicating that it has been corrupted or tampered with.

For more step-by-step instructions, see [How to Configure a TLS Inspection Policy for Outbound TLS Inspection](#) and [How to Configure Outbound TLS Inspection](#).

Inbound TLS Inspection

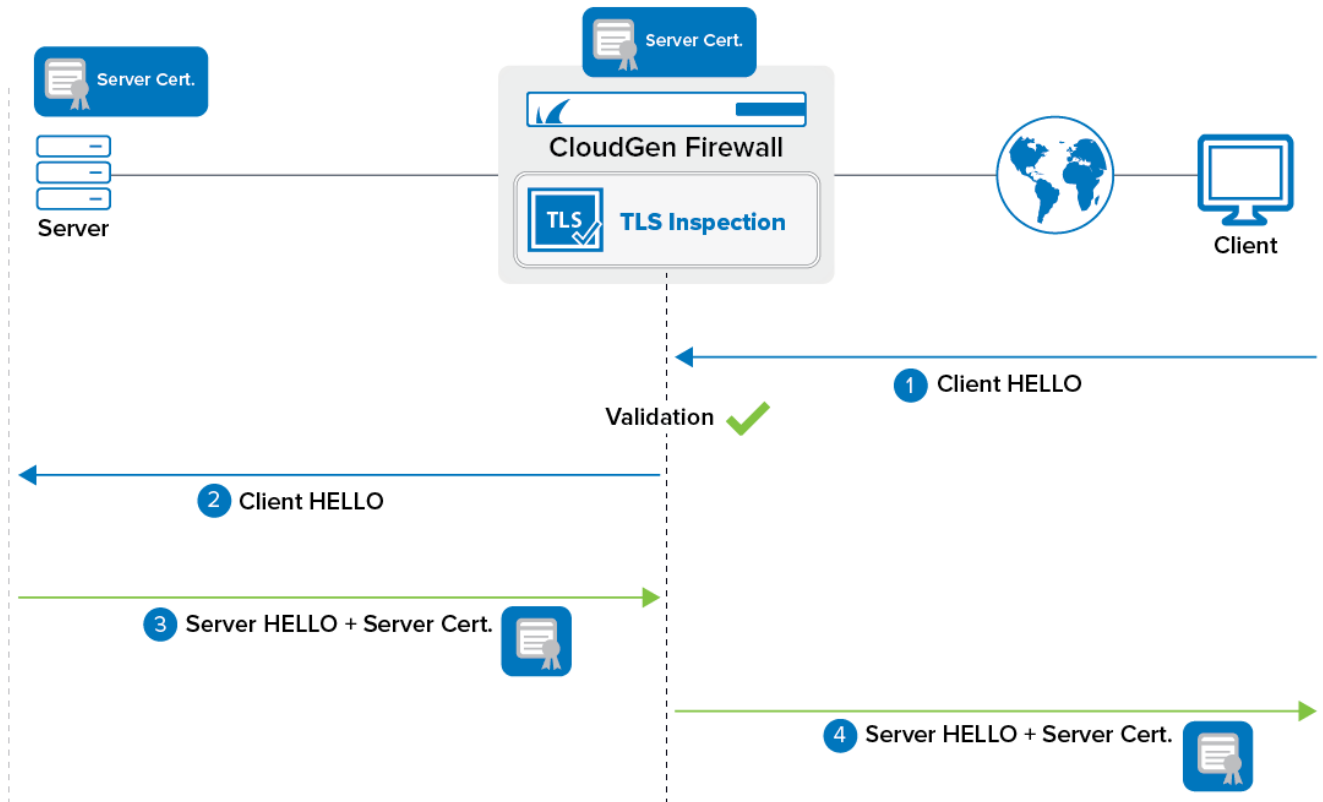


Inbound TLS Inspection is used to terminate the TLS connection of external clients accessing internal servers that are protected by the firewall. The firewall uses the server certificate to negotiate the TLS session. On the firewall, the IPS or Virus Scanner can then scan the traffic before it is forwarded via the TLS connection to the internal server. This means that the client cannot distinguish between the actual server and the firewall when connecting. Half-side encryption is not supported. It is also not possible to use client certificate authentication when using inbound TLS Inspection.

Only RSA- and ECDSA-based server certificates and private keys are supported for imports to the Certificate Store. Therefore, only RSA- and ECDSA-based ciphers can be used, reduced by ciphers with weak temporary/ephemeral key exchanges.

For more information, see [How to Create a TLS Inspection Policy for Inbound TLS Inspection](#) and [How to Configure Inbound SSL Inspection](#).

Inbound TLS Inspection Process



1. Client sends client HELLO to the firewall. This includes supported TLS versions and ciphers.
2. The firewall sends a client HELLO to the server.
3. The server replies with a server HELLO including the server certificate.
4. The firewall sends a server HELLO to the client using the server certificate configured in the TLS Inspection policy. The certificate on the server and the certificate on the firewall must be the same.

Troubleshooting

If a TLS error log message 'SSL_ERROR_SYSCALL: Connection reset by peer' occurs, try the following workaround:

- Configure an access rule for the affected servers in Firewall Admin. In the **Edit Rule** window, select **Advanced**. In the **TCP Policy** section, set **Enable TCP Timestamp stripping** to **Yes**.

If a TLS connection fails with 'ssl3_get_client_hello:no shared cipher', the server's certificate authentication/key exchange type (RSA, DSA, and DH) will not fit that of the selected cipher. Especially for Inbound SSL sessions, there is no support for temporary/ephemeral weak key exchanges. This is true for almost all old export-restricted ciphers, likewise resulting in the error

message above regardless of whether the cipher was configured correctly.

If the **Revocation Check Timeout** is set to a value > ~10 sec. and a response timeout occurs, some *.google.com-server provides no content in the HTTP response. In this case, set the **Revocation Check Timeout** to less than ~10 sec.

If a TLS error log message 'OpenSSL library: error:0A000076:SSL routines::no suitable signature algorithm' occurs, the security level (default=1 in OpenSSL 3.x) overrules the cipher configuration. Solution: to decrease the security level, add "@SECLEVEL=0" at the end of the cipher list.

If a TLS error log message 'OpenSSL library: error:0A000438:SSL routines::tlsv1 alert internal error ...', occurs, the minimum TLS version forbids TLS session establishment. In this case, create a new SSL Inspection policy in Firewall Admin and add "@SECLEVEL=0" at the end of the cipher list to decrease the OpenSSL security level. Then change the minimum TLS version in the bit field manually in the *active.fwrule* file:

- sslConfig= {160260} for SSLv3.0
- sslConfig= {160264} for TLSv1.0
- sslConfig= {160272} for TLSv1.1

TLS Inspection Policies and the Special Policy "Default"

The policy **Default** is synonymous with the legacy SSL Inspection policy used before CloudGen Firewall version 7.2. It refers to all groups in **Certificate Validation Policies** being set to **Pass Error to Client**. For the revocation check, the configuration in **Firewall > Security Policy > Advanced** (next to **Enable SSL Inspection**) > **CRL Error Policy**, and in **Security Policy > Enable CRL Checks** (next to **Trusted Root Certificates**) is used.

- **CRL Error Policy** set to **Fail** invalidates the client-side certificate by modifying the issuer to force a browser warning page.
- **CRL Error Policy** set to **Ignore** hides a CRL/OCSP revocation check error from the client.
- **Enable CRL Checks** switches the CRL/OCSP check to **on** or **off** for all forwarding rules that use **Default**.

If a TLS Inspection Policy is configured and assigned to forwarding rules, the selected *Certificate Validation Policies* and the CRL/OCSP-related configuration in the *Revocation Check Error Policy* are used.

- **Enable Revocation Check** switches the CRL/OCSP check to **on** or **off** for all forwarding rules that use this policy.
- **Action on Revocation Check Error** set to **Fail Open** hides a CRL/OCSP revocation check error from the client.
- **Action on Revocation Check Error** set to **Fail Close** terminates the TLS session.

CloudGen Firewall Initial Start-Up and Root Certificate

The CloudGen Firewall allows TLS Inspection without having an explicit root certificate configured.

This is considered Demo Mode only.

The admin must consider the following: the RSA key and the certificate (CN=Barracuda Networks AG) are created if no explicit root certificate is configured. However, they are available only until the next boxfw process restarts, are insecure (it is a 512-bit key), and cannot be exported for use as clients as a trusted anchor.

An explicit root certificate must be created/configured to ensure a productive system.

Figures

1. tls_inspection_overview.png
2. SSL_block_page.png
3. ciphers.png
4. tls_inspection_out.png
5. tls_outbound_dyn.png
6. tls_inspection_in.png
7. tls_inbound_dyn.png

© Barracuda Networks Inc., 2024 The information contained within this document is confidential and proprietary to Barracuda Networks Inc. No portion of this document may be copied, distributed, publicized or used for other than internal documentary purposes without the written consent of an official representative of Barracuda Networks Inc. All specifications are subject to change without notice. Barracuda Networks Inc. assumes no responsibility for any inaccuracies in this document. Barracuda Networks Inc. reserves the right to change, modify, transfer, or otherwise revise this publication without notice.