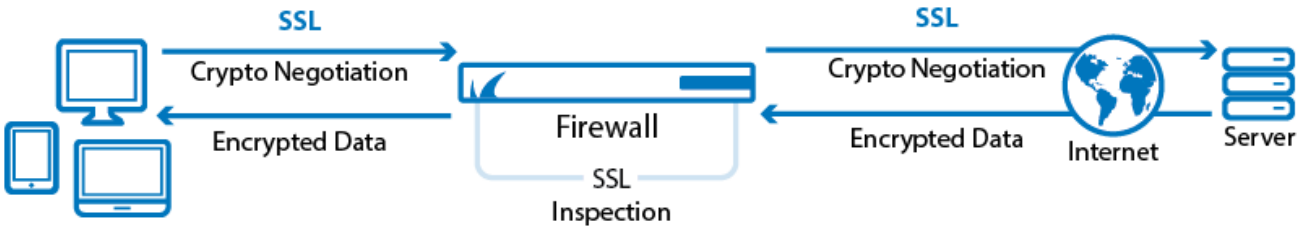
SSL Inspection in the Firewall

https://campus.barracuda.com/doc/73719887/

SSL Inspection decrypts both SSL and 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 Using SSL Inspection allows the admin to enforce SSL/TLS security at the firewall by blocking outdated ciphers or refusing connections attempting to use outdated SSL versions. For outbound SSL Inspection, the firewall can also handle SSL validation errors, depending on the SSL error policy assigned to the matching access rule of the SSL/TLS session. SSL Inspection is supported for Pass, Map, and Dst NAT access rules. Not supported are SSL connections that require client certificate authentication.

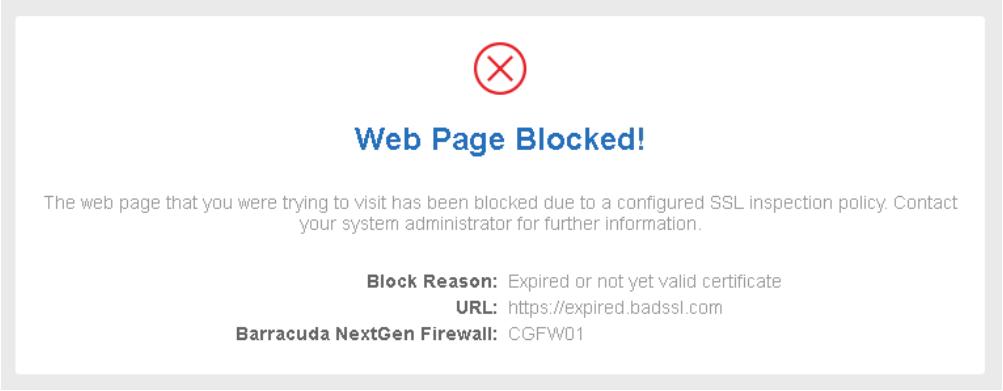


Enforce Ciphers and Minimum SSL/TLS Version

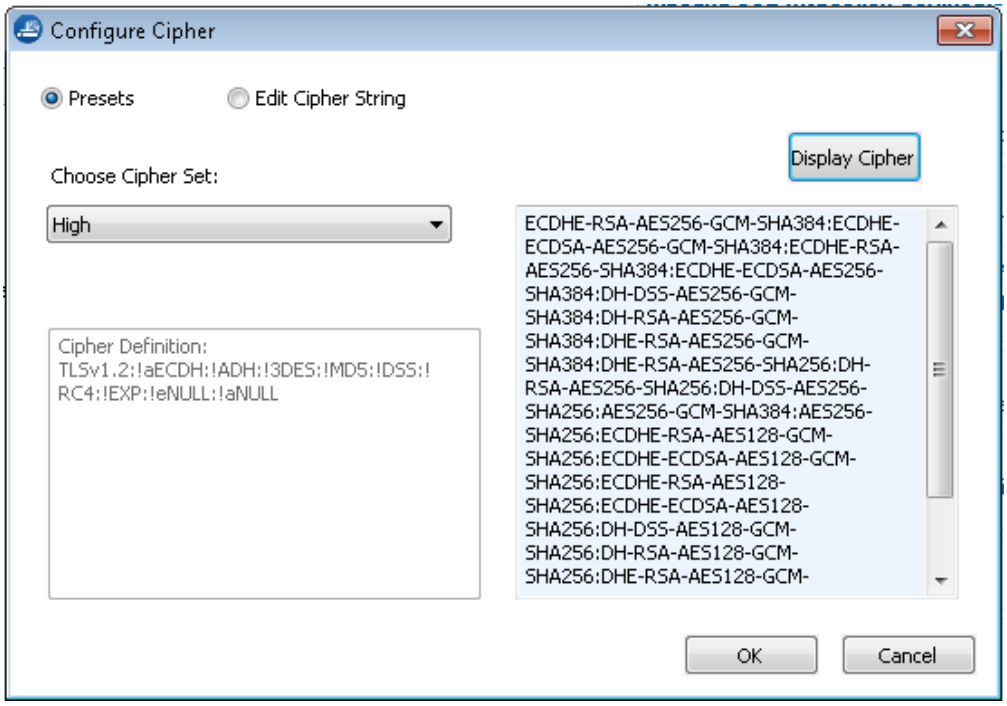
The ciphers and the SSL/TLS version for the SSL or TLS connection are negotiated between the client and the firewall and the firewall and the server. The negotiated settings for the SSL/TLS connection between the firewall and the client and the connection between the firewall and the server always strive to use the most secure cipher and TLS/SSL version possible, which can result in different encryption settings for each connection. For example: an SSL connection from a client that only supports SSLv3, while the server on the other side of the firewall supports TLS 1.2, will result in an SSLv3 connection to the firewall and a TLS 1.2 connection from the firewall to the server, if the settings of the SSL Inspection policy allow these connections.

SSL Inspection on the firewall allows the admin to define the minimum supported SSL/TLS version and a suite of allowed ciphers on a per-access-rule basis. By using different SSL 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 SSL/TLS connections. Three preset cipher sets (HIGH, MEDIUM, and LOW) are available. The ciphers included in the presets are defined by the openSSL 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 SSL 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 openSSL cipher string (LOW, MEDIUM, and HIGH) by appending your own openSSL-compatible string. Use the syntax compatible with the OpenSSL version used by the firewall. The syntax of the cipher string may have to be changed if the underlying openSSL version has changed during the update. The openSSL version is listed in the Firewall/SSL log when log verbosity is set to **info** in the advanced SSL Inspection setting in the Security Policy settings.



Certificate Revocation Checks

Since clients behind a firewall using SSL Inspection cannot complete their own revocation check, the firewall handles the revocation check during the SSL 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

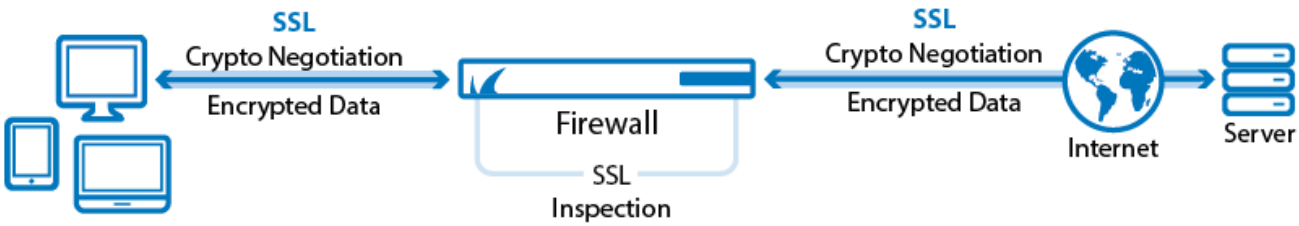
OCSP stapling is an offline check; for OCSP and CRL, the firewall must be online. The first revocation check that returns a valid result is used. To reduce the load on the OCSP or CRL servers, the revocation check is done either once per 24h period, or immediately after an SSL Inspection configuration 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 SSL Inspection > Advanced**). Depending on the fail close or fail open policies, SSL connections are either blocked or allowed when the revocation check fails due to operational reasons, such as an unreachable OCSP server.

Exceptions from SSL Inspection

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

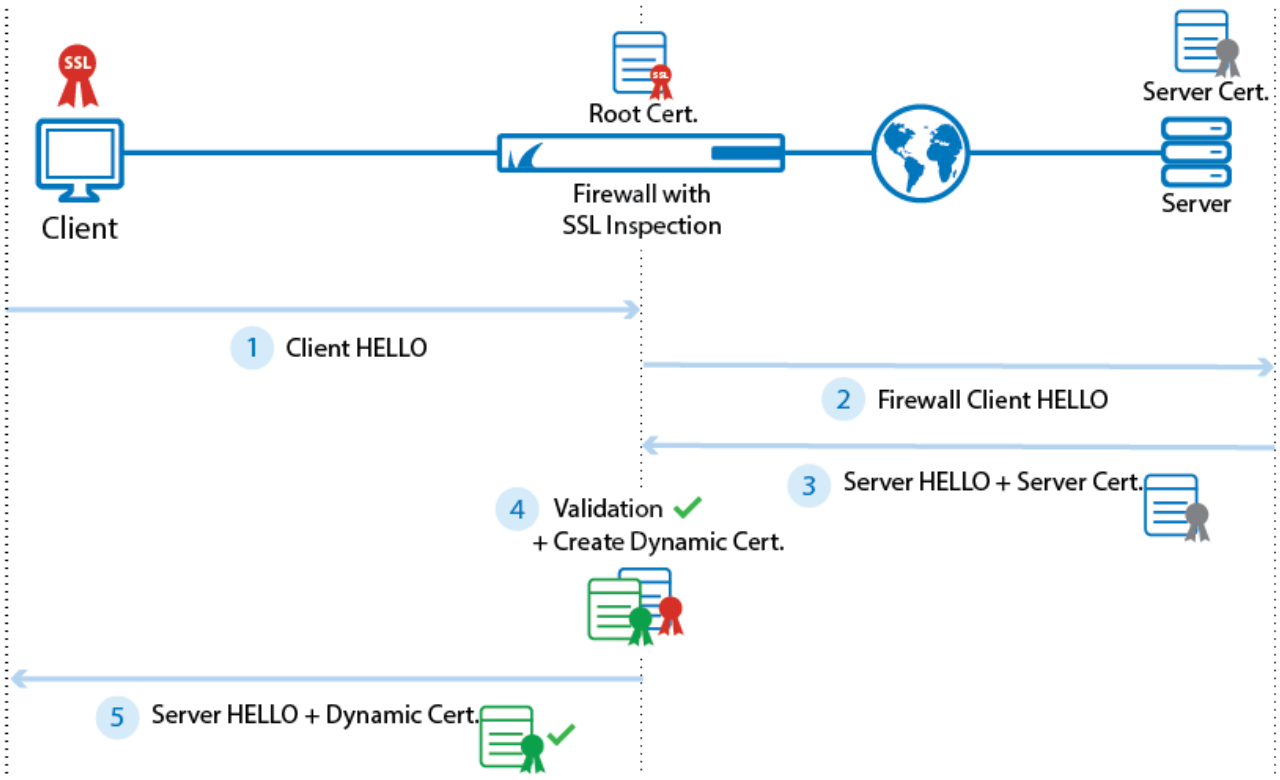
- (outbound SSL Inspection only) Applications marked as **not interceptable** in the application object properties. Go to the application object on the **Forwarding Rules > Applications** page to check if an application is interceptable.
- Domains whitelisted in the Security Policy settings.
- URL categories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

Outbound SSL Inspection



Outbound SSL Interception inspects SSL and TLS connections from clients behind the firewall to the Internet. The firewall decrypts the SSL 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 SSL Inspection root certificate. The traffic is re-encrypted and forwarded to the destination. For the client to accept the intercepted SSL session with the dynamic self-signed certificate, the SSL Inspection root certificate must be placed in the Trusted Root Certificates store on every client behind the firewall.

Outbound SSL and TLS Inspection Process



1. Client sends client HELLO to server. This includes supported TLS/SSL versions and ciphers.
2. Connection is intercepted (blocked) by the firewall. Firewall sends client HELLO to the server using the ciphers and supported SSL/TLS versions in the SSL Inspection policy.
3. Server replies with the server HELLO, which includes the servers 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 sever 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 client using the dynamic certificate.
7. The client trusts the root certificate of the firewall and the connection is established.

SSL Policy Validation Errors

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

- **Pass Error to Client (default)** – A purposefully invalid SSL certificate is generated for the client, causing an error message on the client.
- **Hide Error from Client** – The client receives a valid SSL certificate, even if the SSL or TLS connection causes an SSL error on the firewall.

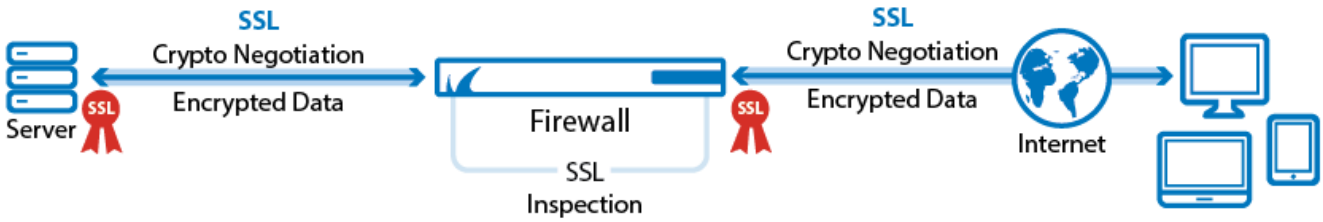
- **Block** – The connection is blocked on the firewall. HTTPS connections are redirected to a block page.

The following SSL 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 an SSL Inspection Policy for Outbound SSL Inspection and How to Configure Outbound SSL Inspection.

Inbound SSL Inspection



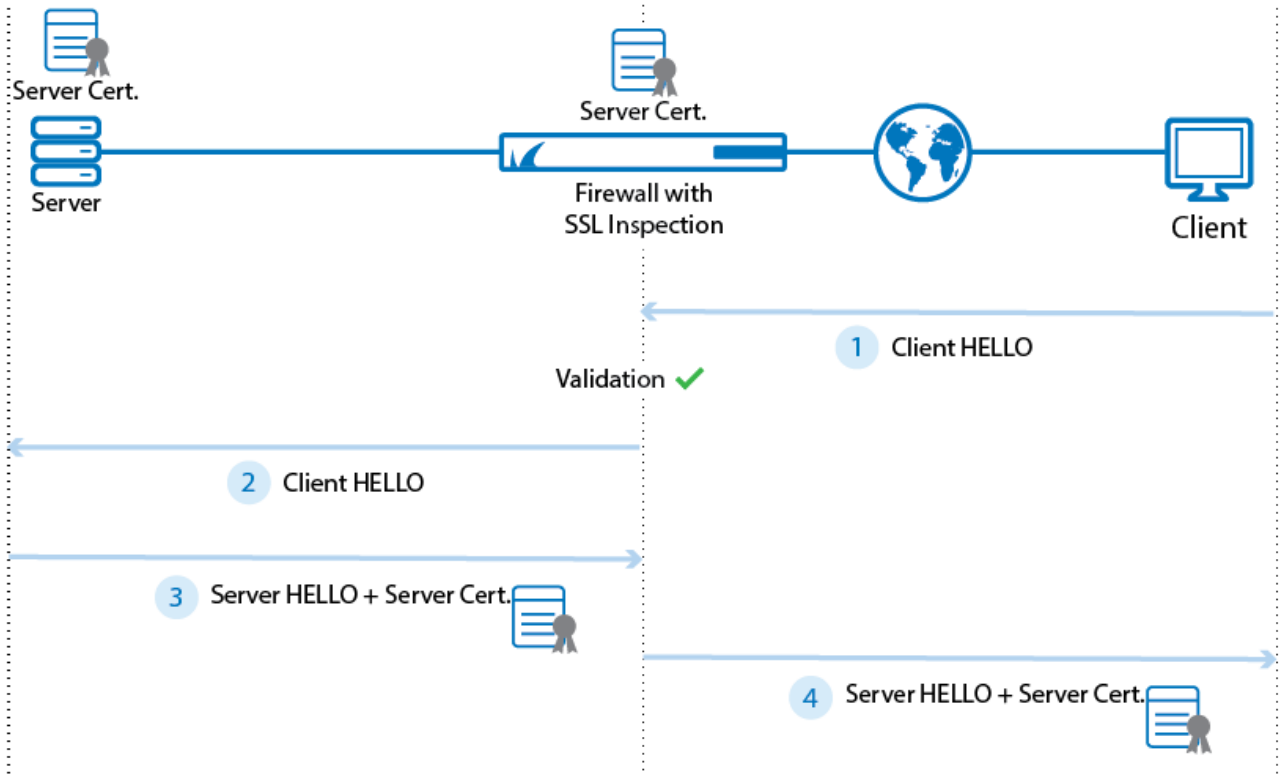
Inbound SSL Inspection is used to terminate the SSL connection of external clients accessing internal servers that are protected by the firewall. The firewall uses the server certificate to negotiate the SSL or TLS session. On the firewall, the IPS or Virus Scanner can then scan the traffic before it is forwarded via the SSL 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 SSL Inspection.

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

For more information, see How to Create an SSL Inspection Policy for Inbound SSL Inspection and How

to Configure Inbound SSL Inspection.

Inbound SSL and TLS Inspection Process



1. Client sends client HELLO to the firewall. This includes supported TLS/SSL 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 SSL Inspection policy. The certificate on the server and the certificate on the firewall must be the same.

Troubleshooting

If an SSL error log message '`SSL_ERROR_SYSCALL: Connection reset by peer`' occurs, try 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 an SSL 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.

SSL 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 an SSL 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* is 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 SSL session.

CloudGen Firewall Initial Start-Up and Root Certificate

The CloudGen Firewall allows SSL 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. ssl_inspection.png
2. SSl_Inspection_block_page.png
3. ciphers.png
4. ssl_inspection_outbound.png
5. ssl_outbound_dynamic01-01.png
6. ssl_inspection_inbound.png
7. ssl_inbound_dynamic01-01.png