BLUE COAT

ProxySG Appliance/ SGOS

Blue Coat Security First Steps

Solution for Controlling HTTPS

Security Empowers Business

SGOS 6.5

Legal Notice

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

www.symantec.com

10/23/2017

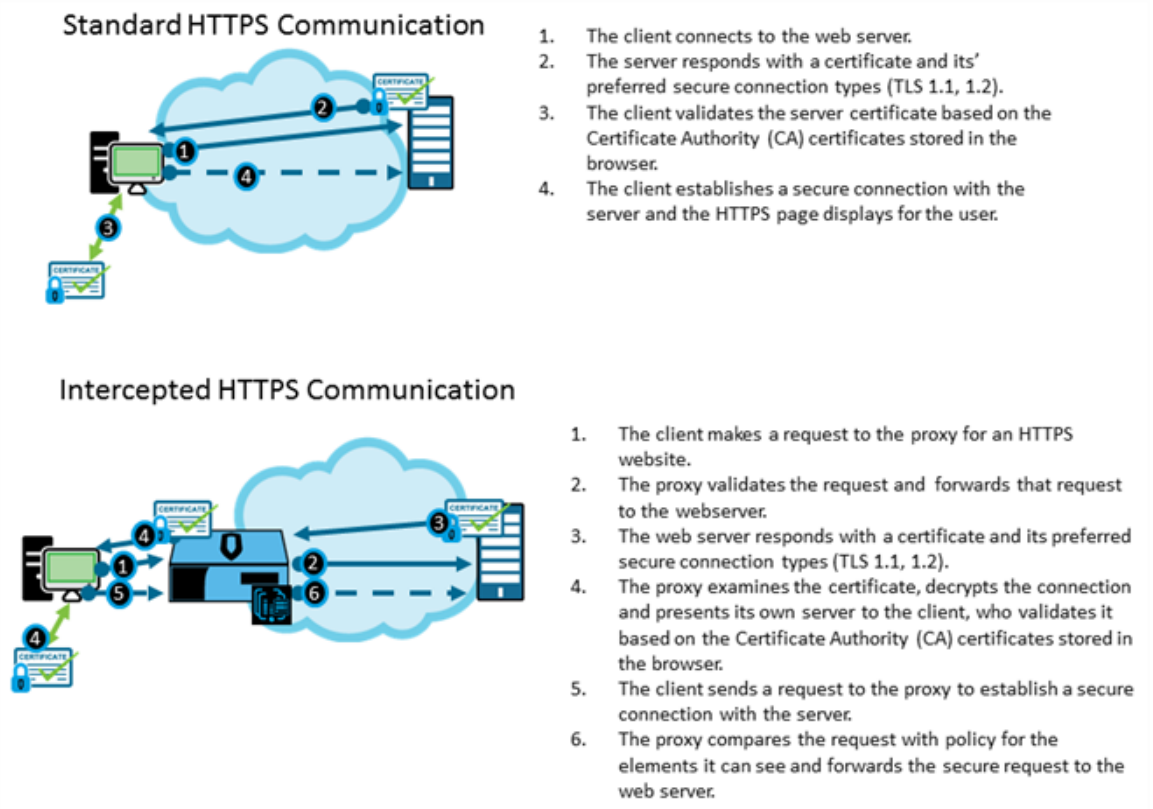
Contents

Solution: Control HTTPS Traffic

A large percentage of Internet traffic uses HTTPS to secure client-to-server communication. HTTPS negotiation is based on a trust relationship between the client (such as a web browser) and the server (the web server hosting the HTTPS page). That trust relationship is established when the client first connects to the server. At that point, the server identifies itself with a certificate that has been signed by an entity (a Certificate Authority) that the browser trusts. After the client and server establish a trust relationship, they agree on the level and type of security (typically TLS 1.1 or higher) that will be used to secure the connection.

In the context of controlling your users' Internet traffic with a ProxySG appliance, HTTPS traffic presents a few challenges. With a simple set of policies used to control HTTP traffic, the appliance can only identify the unencrypted portions of the exchange between the client and the server in an HTTPS request and response scenario. The unencrypted data includes the client IP address, the port number used to establish the connection, the server's certificate, and some connection details, depending on the type of proxy deployment. In an explicit proxy deployment, policy can identify the server's hostname. In a transparent proxy deployment, policy can only see the IP address of the server. With this limited amount of information, controlling client requests based on elements such as the URL path fail.

To affect policy further, the ProxySG appliance decrypts HTTPS traffic in transit, so that it can apply policy to it.This process is known as *SSL interception*. SSL interception works by having the client establish a trust relationship with the appliance, which can then enforce policy such as simple allow/deny actions based on the entire URL path of a request, or even advanced elements such as authentication, access logging, and user notification. The connection is re-encrypted before the proxy sends the request to the web server requested by the client, then the process happens in reverse for the server's response to the client.



Steps

1. Make sure your ProxySG is set up properly for SSL. See "Verify Your ProxySG Setup for SSL" below.
2. Create an issuer keyring that uses either a self-signed or CA-signed certificate for SSL decryption (the CA-signed page contains an example Microsoft® setup; your setup may vary).
3. Configure the appropriate service for your deployment:

   "Configure SSL Proxy Services in an Explicit Deployment" on page 9

   or

   "Configure SSL Proxy Services in a Transparent Deployment" on page 9.
4. "Configure Policy for the SSL Proxy" on page 10.
5. "Verify SSL Traffic Interception" on page 11.
6. To ensure authentication works in a transparent deployment, see "Enable IWA Authentication for SSL Traffic in a Transparent Deployment" in the Authentication solution.

Verify Your ProxySG Setup for SSL

Verify the following before you begin setting up the SSL Proxy to intercept traffic.

- Verify your ProxySG appliance is functional, and can permit a proxied user to access Internet resources.
- Determine if your domain uses certificate signing services, This helps you decide whether they'll be following the self-signed or CA-signed certificate process.If using a CA-signed certificate, have access to your certificate signing authority for your domain.
- Determine whether you are in a transparent or explicit setup.
- Verify the ProxySG is configured with a basic license *and* an SSL license. Go to: **Maintenance** tab > **View** > **Licensing** > **Licensed Component**; and verify **SGOS 6 Proxy Edition** and **SSL**.
- Verify the ProxySG to set to get its time from a reputable and reliable time source. To review your NTP settings on the ProxySG, check Configuration > General > Clock. Any discrepancies between the date and time in certificates created by the ProxySG and the actual time can cause unexpected behavior, as such it is important that the time on the ProxySG be set up correctly before proceeding.

Next Step: "Create a Keyring" below

Create a Keyring

In order for the ProxySG appliance to decrypt SSL traffic, a keyring must be created. Keyrings are containers for SSL certificates on the appliance, and can be used to manage self-signed or CA-signed certificates.

1. Select **Configuration > SSL > Keyrings > Create**.
2. For **Keyring Name**, enter a descriptive name for the keyring.

3. Select **Show key pair**, based on your security policy.
4. Leave the key size at 2048 bits.
5. Click **OK** and **Apply** to save your changes.

Next Step: "Create a Self-Signed Certificate" below or "Create a CA-Signed Certificate" below

Create a Self-Signed Certificate

Create a self-signed certificate if you don't want to use the Default certificate. If you do use the Default, you can skip this step.

1. In the Keyrings tab, **Edit** the newly created keyring (see "Create a Keyring" on the previous page).
2. Under **Certificate**, click **Create**. The Create Certificate form displays.
3. Fill in the certificate details.
   - The **Common Name***must* match the ProxySG name or IP address that the client expects (the name for preference).
   - Make note of the **Challenge** you create.
4. Click **OK** and then **Close**. For the keyring you created, **Yes** appears in the Certificate column, indicating a certificate is part of the keyring.
5. **Apply** your changes.

Next Step: "Configure SSL Proxy Services in an Explicit Deployment" on page 9 or "Configure SSL Proxy Services in a Transparent Deployment" on page 9

Create a CA-Signed Certificate

Generate a CA-signed certificate for your SSL interception keyring.

Select **Configuration > General > Clock** to confirm correct time configuration and NTP settings. Because SSL certificates include a date and time component, configure an NTP server to prevent potential problems.

The following steps use Microsoft PKI server as an example. Where necessary, refer to your server/administration documentation for steps specific to your environment.

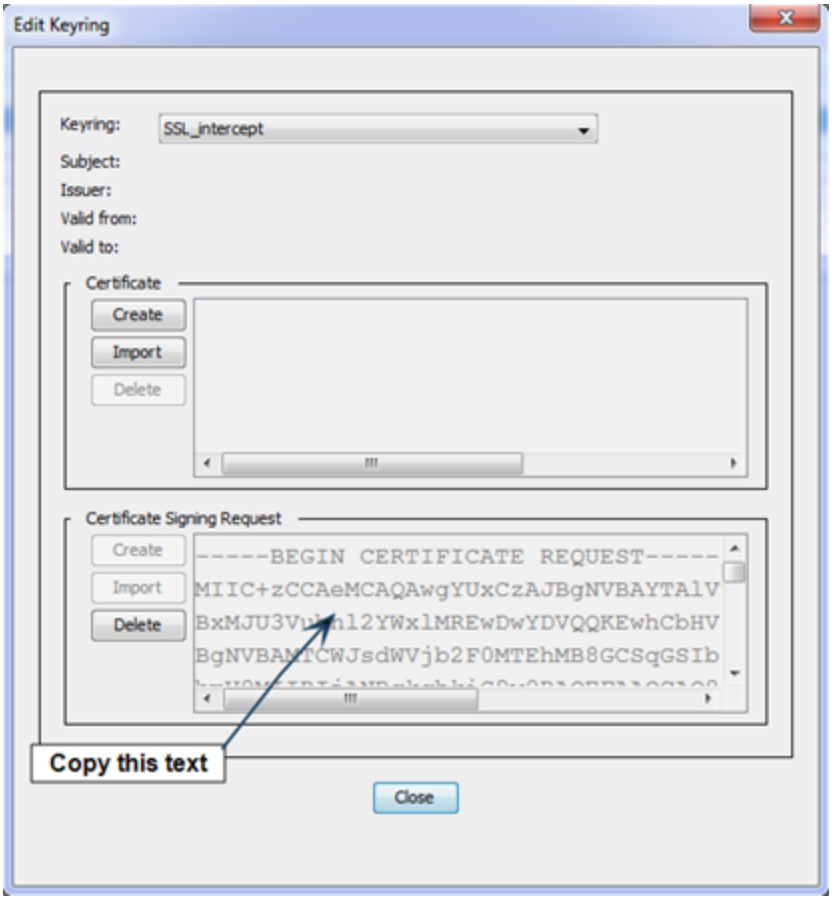
1 - Create a Certificate Signing Request

Generate a certificate signing request (CSR) on the ProxySG appliance. You will use the CSR to request the certificate from a certificate signing authority.

1. In the ProxySG Management Console, select **Configuration > SSL > Keyrings**.
2. Select the interception keyring created in "Create a Keyring" on the previous page. Then, click **Edit**.
3. Under **Certificate Signing Request,**click **Create**.
4. Complete the fields as appropriate. Set the **Common Name** to the single hostname (resolvable via DNS) of the ProxySG. Click **OK**, click **Close**, and then click **Apply**.

If the DNS name of the ProxySG appliance is not specified for **Common Name**, the web browser will return a not-trusted warning.

5. Edit the keyring. The Certificate Signing Request field displays the contents of the CSR.
   a. Copy this text to the clipboard (including the `BEGIN` and `END` text).



   b. Save it in a file and give it a meaningful name.
   c. Click **Close**.

2 - Request and Download Certificates

Use the CSR to request and download the following from a trusted certificate signing authority:

- a subordinate/intermediate certificate for ProxySG interception
- (if not already created and downloaded) the root CA certificate

This example uses Microsoft PKI as the certificate signing authority. Microsoft PKI services require that the user account is a domain administrator.

1. In Internet Explorer, go to the URL of the Microsoft Active Directory Certificate Services server you use to contact the certificate signing authority (typically http://<*windows_server_IP_address*>/certsrv/).
2. Click **Request a certificate > Advanced Certificate request**.

Be sure to select **Advanced certificate request**; **User Certificate** refers to an individual user certificate.

3. Download the subordinate/intermediate certificate:
   a. Select **Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request using a base-64-encoded PKCS #7 file.**
   b. Copy the contents of the *.csr file you generated in Create a Certificate Signing Request.
   c. Paste the contents into the **Base-64-encoded certificate request (CMC or PKCS#10 or PKCS#7)** field.
   d. Select **Certificate Template > Subordinate Certification Authority**, and then click **Submit**.
   e. Select **Base 64 encoded**, and then click **Download Certificate**. Save the certificate with a meaningful name. The subordinate/intermediate certificate is saved.
4. (If needed) Download the root certificate:
   a. Click **Home**.
   b. Click **Download a CA certificate, certificate chain, or CRL**.
   c. Select the appropriate CA Certificate from the list at the top, select **Base 64** as the encoding method, and click **Download CA certificate**. Save the certificate with a meaningful name. The CA certificate is saved.

3 - Install Certificates

Install the certificate(s) on the ProxySG appliance:

1. In the ProxySG Management Console, select **Configuration > SSL > Keyrings**.
2. Select the interception keyring created in "Create a Keyring" on page 5. Then, click **Edit**.
3. Select **Certificate > Import**.
4. Copy and paste the contents of the interception keyring into the **Import Certificate** dialog. Then, click **OK**, **Close**, and **Apply** to save your changes.

   If you import an incorrect certificate, clicking **Apply** displays an error message such as “The private key in the certificate '*<certificate_name>*' does not match the one in the keyring.”

5. Add the ProxySG interception certificate to the list of CA certificates on the ProxySG:
   a. Go to **Configuration > SSL > CA Certificates**.
   b. Click **Import**.
   c. Name the certificate, and then copy and paste the contents of the subordinate/intermediate certificate you downloaded in Request and Download Certificates.
   d. Click **OK** and then **Apply**. The CA Certificates list displays the certificate.
6. (If you have already imported the root CA certificate, skip this step.) Add the root CA certificate to the list of CA certificates on the ProxySG:
   a. Click **Import** on the **CA Certificates** tab.
   b. Name the certificate, and then copy and paste the contents of the root CA certificate you downloaded in Request and Download Certificates.
   c. Click **OK** and then **Apply**. The CA Certificates list displays the certificate.
7. (If you have already added the root CA certificate to the browser-trusted CA, skip this step.) Add the new certificates as browser-trusted CAs:
   a. Select the **CA Certificates Lists** tab.
   b. Select **browser-trusted**, and then click **Edit**.
   c. Select the certificate(s) you imported and click **Add** to move them to the right column.
   d. Click **OK**, and then click **Apply**.

8. Set the SSL Proxy to use the new keyring:
    a. Select **Configuration > Proxy Settings > SSL Proxy**.
    b. Under General Settings, set **Issuer Keyring** to the new keyring.
    c. Click **Apply**. The SSL Proxy now uses the new keyring.

Next Step: "Configure SSL Proxy Services in an Explicit Deployment" below or "Configure SSL Proxy Services in a Transparent Deployment" below

Configure SSL Proxy Services in an Explicit Deployment

In an explicit deployment, you need to intercept the Explicit HTTP service, and make sure it has Protocol Detect enabled.

Follow these steps in an explicit deployment to intercept and manage SSL traffic.

1. Select **Configuration > Services > Proxy Services > Standard > Explicit HTTP**.
2. Click **Edit Service**. The Management Console displays the Edit Service window.
3. In the **Proxy settings** section:

- **Proxy**: **HTTP**
- **Detect Protocol**: checked; this allows the service to see the host and path of the request, so the HTTP proxy can detect HTTPS traffic and hand it off to the internal SSL Proxy engine.

5. In the **Application Delivery Network Settings** section:

- **Enable ADN**: unchecked

6. In the **Listeners** section:

- **Port range**: **8080**
- **Action**: **Intercept**

6. Select **OK** as required to close the windows.
7. Select **Apply** to confirm your changes.

Next Step: "Configure Policy for the SSL Proxy" on the facing page

Configure SSL Proxy Services in a Transparent Deployment

In a transparent deployment, you need to intercept the HTTPS service, and make sure it has Protocol Detect enabled. Depending on how user traffic reaches your ProxySG, the steps to configure the service listeners vary.

1. Select **Configuration > Services > Proxy Services > Standard > HTTPS** .
2. Click **Edit Service**. The **Edit Service** window displays.
3. In the **Proxy settings** section:
    - **Proxy**: **HTTPS**
    - **Detect Protocol**: checked; this allows the service to see the host and path of the request, so the HTTP proxy can detect HTTPS traffic and hand it off to the internal SSL Proxy engine.
4. In the **Listeners** section:

- **Action: Intercept**
- **Port range: 443**. This is set at 443 for transparent.

4. Click **OK** as required to close the windows.
5. Click **Apply** to confirm your changes.

Next Step: "Configure Policy for the SSL Proxy" below

Configure Policy for the SSL Proxy

SSL interception and access rules, including server certificate validation, are configured through policy, created in the VPM or written in CPL.

- Use the *SSL Intercept Layer* to configure SSL interception.
- With SSL interception in place, all standard web access layer rules and actions will apply to SSL-intercepted requests. For example, the web access layer can be set to match the Destination, and further refined as required.
- Use the *SSL Access Layer* to control other aspects of SSL communication, such as server certificate validation, perform actions on non-intercepted sites based on the SSL certificate presented by secure sites, and control the SSL versions to support.
- Optionally, set a Web Access Layer to match the Destination, and refine as required.

Typically, you will want to intercept HTTPS (secure Internet) traffic. Once the traffic has been intercepted, you can perform many additional tasks.

The Default proxy policy on **Configuration tab > Policy Options > Default Proxy Policy** supersedes all other policy. It is set to **Deny**. Typically, you will switch this to **Allow**, then define denials through policy, with a last deny policy as a catch all.

Install Basic HTTPS Interception Policy

1. Open the Visual Policy Manager (select **Configuration > Policy > Visual Policy Manager > Launch**).
2. Select **Policy > Add SSL Intercept Layer**, name the layer, and click **OK**.
3. Right-click in the new rule's **Action** column, and select **Set > New > Enable SSL Interception**.
4. In the Add Enable SSL Interception Object window:
   a. Verify **Enable HTTPS interception** is selected.
   b. Check **Issuer Keyring**.
   c. From the drop-down menu, select the SSL keyring you created for SSL interception.
   d. Click **OK**.
   e. Click **OK**.
5. Click **Install policy** on the **VPM** window to save the SSL policy. You will see a confirmation message. Close the window.

Make sure you have set up SSL proxy services, for transparent or explicit environments, as required.

Next Step: "Verify SSL Traffic Interception" below

Verify SSL Traffic Interception

Here are a few ways to monitor the performance of the SSL Proxy.

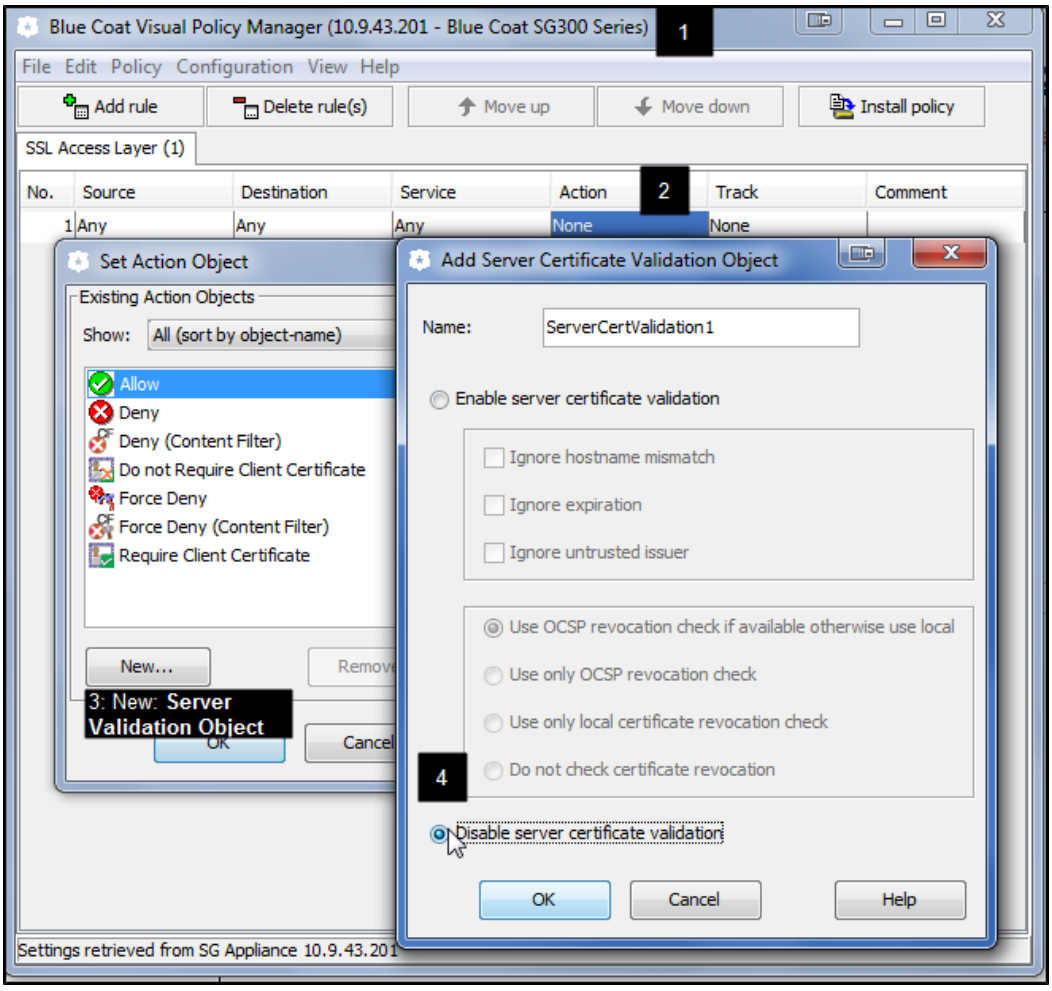
- Open a browser. Without SSL interception, the browser shows a lock icon in the address or status bar (depending on the browser). Clicking that lock shows who the issuer is, and that they're trusted. With SSL interception in place and working, the ProxySG's DNS name or IP address, depending on the configuration, is listed as the issuer.
- In your browser, look at the certificate info and verify it matches what you expect, including the IP address of your ProxySG.
- On the ProxySG,go to **Statistics > Active Sessions**, and verify the HTTPS forward proxy is listed.
- Look for bandwidth gain in the **Statistics > Traffic Details** reports.

SSL Proxy Troubleshooting

Why does the ProxySG not trust allowed sites?

Problem: Users are frequently having connections blocked to allowed sites.

Resolution: By default, the ProxySG does not trust invalid certificates. In the real world, there are more untrusted certificates than trust-able ones. If your users are frequently having connections to allowed sites blocked, you can fix the problem by disabling server certificate validation. Without this policy rule, users can get places, but are blocked due to Certificate Not Trusted errors.



1. In the VPM, add an SSL Access layer (if you don't already have one) or add a rule in the existing SSL Access layer.
2. In the row for the new rule, right-click in the Action column, and select **Set**.

3. In the Set Action Object window, select **New > Set Server Certificate Validation**.
4. In the Add Server Certificate Validation Object window, check **Disable server certificate validation**, then click **OK**.
5. In the VPM window, click **Install policy**.