riverbed
Think fast.™

SOLUTION GUIDE

Creating Proxy Certificates for Riverbed® Steelhead® Appliances using a Microsoft® Certificate Authority

Solution Guide

Version 1.0
September 2013

Contents

PREFACE

Welcome to the *Creating Certificates for Riverbed Steelhead Appliances using a Microsoft Certificate Server Guide*. Read this preface for an overview of the information provided in this guide and contact information. This preface includes the following sections:

- About This Guide
- Contacting Riverbed

About This Guide

Modern networks frequently use the TLS/SSL protocol standards to encrypt network traffic. When using encryption, if two users download the same document, a completely different set of bits is sent across the wire. This happens as each user's session is encrypted with a different secret key that is setup as part of the encrypted connection. Riverbed Steelhead appliances can transparently decrypt, optimize, and re-encrypt TLS/ SSL traffic while maintaining end-to-end secure encryption, allowing the Steelhead appliances to provide highly effective WAN optimization for encrypted network traffic.

To optimize encrypted traffic, one or more certificates must be created that contain the private key from a trusted certificate authority (CA) and installed on the server side Steelhead. Many customer networks use Microsoft Active Directory with one or more servers acting in the role of a certificate server. *Creating Certificates for Riverbed Steelhead Appliances using a Microsoft Certificate Server* guides you through the process of determining what information you need and then to create certificates using a Microsoft certificate server.

This guide is intended to be used together with the following documentation:

- *Steelhead Deployment Guide - Protocols* – The chapter on Chapter on SSL Deployments includes information on SSL and Steelhead appliances.
- *Solution Guide - Optimizing Microsoft 2008 R2 SSL-based traffic* –This document describes creating certificates for your intranet applications.

This guide includes information relevant to the following products:

- Riverbed Steelhead appliance (Steelhead appliance)
- Riverbed Steelhead CX appliance (Steelhead CX)
- Riverbed Virtual Steelhead appliances (Virtual Steelhead appliance) for Hyper-V and VMware
- Microsoft Windows 2008R2 SP1

Audience

This guide is written for administrators of Steelhead appliances who are familiar with SSL, certificates, and have the authority to create or request the creation and issuance of certificates using a Microsoft certificate server in their network.

You must also be familiar with the Steelhead Appliance Management Console. For details, see the *Steelhead Appliance Management Console User's Guide*.

For more details on the Riverbed Steelhead product family, see http://www.riverbed.com/products-solutions/products/wan-optimization-steelhead/

Contacting Riverbed

This section describes how to contact Riverbed.

Internet

You can learn about Riverbed products through the company Web site: http://www.riverbed.com.

Technical Support

If you have problems installing, using, or replacing Riverbed products, contact Riverbed Support or your channel partner who provides support. To contact Riverbed Support, open a trouble ticket by calling 1-888-RVBD-TAC (1-888-782-3822) in the United States and Canada or +1 415 247 7381 outside the United States. You can also go to https://support.riverbed.com.

Professional Services

Riverbed has a staff of professionals who can help you with installation, provisioning, network redesign, project management, custom designs, consolidation project design, and custom coded solutions. To contact Riverbed Professional Services, email proserve@riverbed.com or go to http://www.riverbed.com/us/products/professional_services/.

Chapter 1 Solution Overview

This chapter provides an overview of how SSL optimization works with Steelhead appliances and includes the following sections:

- Introduction to SSL Optimization with Steelhead Appliances
- Requirements

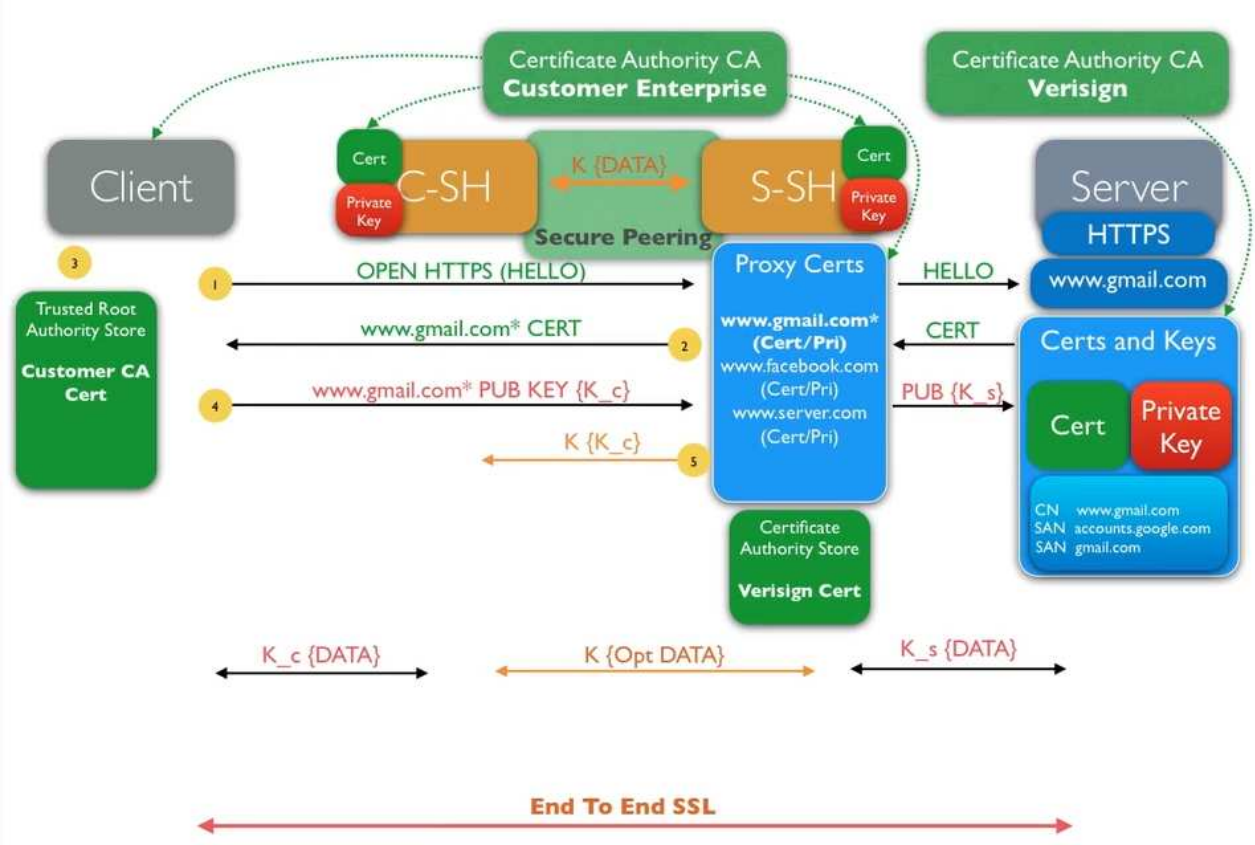
Introduction to SSL Optimization with Steelhead Appliances

The Riverbed method for optimizing SSL insures that traffic is encrypted end-to-end using trusted certificates. This section discusses how SSL optimization with Steelhead appliances works. The following discussion assumes the client uses a browser to access https://www.gmail.com.

SSL Optimization Overview

When implementing end-to-end SSL, tthree different SSL sessions are required. The session between the client and the client-side Steelhead appliance (C-SH) labeled K_c {DATA} near the bottom of Figure 1. A second SSL session is established between the client-side and server-side Steelhead appliances (K {Opt DATA}, also called the 'inner channel'), and the third session established between the server-side Steelhead appliance and the host {K_s {DATA}).

Figure 1 – Simplified view of SSL with Steelhead Appliances



The inner channel uses certificates designated when configuring secured peering on the Steelhead appliances. As this session occurs between trusted Steelhead appliances, the certificates can be self-signed and do not need to be trusted by end users or other servers. In other words, when secure peering is enabled, all traffic between the Steelhead appliances can be encrypted regardless of the type of content, client, or server involved with the traffic. SSL secure peering is generally enabled when SSL

optimization is enabled with the Steelhead appliances and is not in scope for this document.

Note: Peering certificates must allow client authentication as well as server authentication and content encryption.

The numbered arrows in Figure 1 outline a simplified view of how the sessions are established.

1. The client browses to https://www.gmail.com.
2. The S-SH sees the OPEN HTTPS request from the client and matches a server name in the URL to a proxy certificate created by you, using a certificate authority trusted by the client. This certificate is installed on the S-SH for the purpose of optimizing www.gmail.com traffic. In effect, the S-SH intercepts this request and does not pass the request to www.gmail.com at this time..

   Note: SAN names on the proxy certificate are not used for matching. The URL must match the Common Name.

3. The S-SH continues the process of building an SSL connection with the client by sending the selected certificate to the client.
4. The client inspects the certificate and verifies the certificate authority is trusted.
5. After the certificate is accepted, the client and S-SH establish a common session key {K_c} that is used to encrypt data between the client and Steelhead appliances.

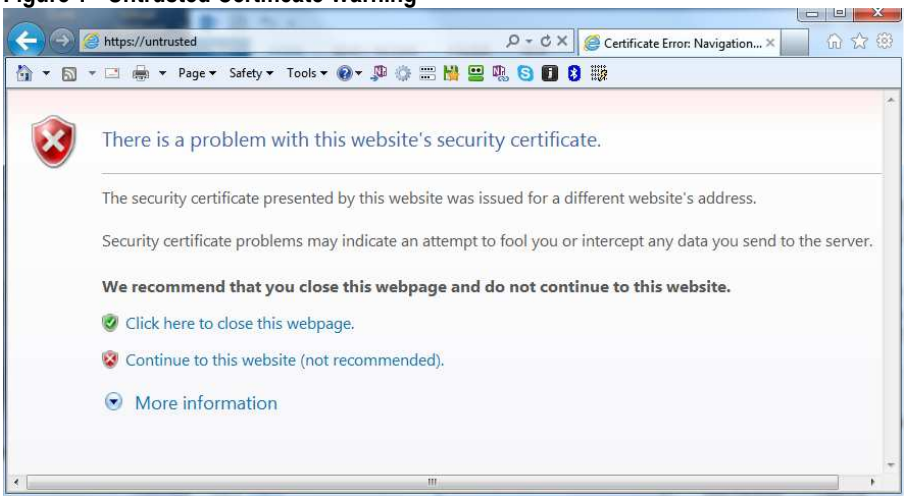
Simultaneously, the S-SH establishes an SSL session {K_s} with the server that responds to www.gmail.com. This session is built exactly as if the client had connected directly with Gmail™ using the certificates provided by Google. These certificates are signed by a well-known certificate authority (represented in Figure 1 by VeriSign).

In this way, communications between the client and Gmail are encrypted end-to-end and optimized, transparently to the client.

Selecting a CA

If at any point the client is presented with a certificate that is not trusted, the client browser or application will present a warning to the user. Figure 2 shows the standard warning used in IE 9. Google Chrome™ and Mozilla Firefox® present different messages, but have the same meaning.

Figure 1 - Untrusted Certificate Warning



As a result, when planning on certificates for use with end-to-end SSL, **in all scenarios,** *the client computer must trust the certificate authority that signs the proxy certificates deployed on the server-side Steelhead appliance.*

Here are some considerations related to the decision about what kind of CA to use:

1. If you own all of the domain names used in your application, you can use a well-known CA, such as VeriSign, to create the certificates.
2. If you want to optimize a service such as the Google Apps™ service, you must create your own certificates that represent the target URLS such as mail.google.com. The CA used for this purpose must be trusted by the client systems. There are two kinds of certificate authorities available to you.
    a. Stand-alone CA: You can create a stand-alone certificate authority using a Microsoft server OS or OpenSSL. In this case, *you must also import that CA root certificate to each computer's Certificate Authority Root store.* This can be done using group policy or other method, but must be done to avoid errors such as those shown in Figure 2.
    b. Enterprise CA: Many organizations that have Active Directory have an enterprise CA. Certificates created by the enterprise CA are automatically trusted by the computers in the forest.

Requirements

The following hardware and software are required for creating certificates for Steelhead appliances using a Microsoft CA:

- Steelhead appliances
- A Microsoft Active Directory® environment with a server deployed as either a stand-alone CA or Enterprise CA
- Administrative access to the Steelhead appliances and the certificate server.

Chapter 2 Preparing to Create Proxy Certificates

This chapter describes how to gather the necessary information to create the required proxy certificates.

Prerequisites

To use the recommended procedure you will need:

- Wireshark – available at http://wireshark.org
- Login credentials on the service you are optimizing, if needed.
- A list of all the URLs in use in your day-to-day operations

---

Overview

In order for the Steelhead appliances to optimize SSL traffic, they must have the capacity to decrypt the SSL traffic coming from an application and then re-encrypt it using keys trusted the clients. If you were optimizing your own website, you could easily obtain these keys as you would own the certificates and could create the necessary certificates containing the private keys used by a certificate internal certificate authority. However, when optimizing cloud services, the provider will not release the private keys, as this would be a serious security breech. Even so, you can decrypt and optimize this traffic using trusted certificates you create. This method of creating your own certificates, often called proxy certificates, for an outside service--such as Google Apps--and deploying them on your own equipment is commonly practiced. Proxy certificates are just like any other certificate you would use for a service, such as a certificate deployed on a web server to provide SSL services.

Sample Scenario

You can use the methods described in this document to setup SSL optimization for most any hosted on premise service. This paper uses Google Apps as an example. The same process would apply to other SaaS services.

Identifying Certificates In Use

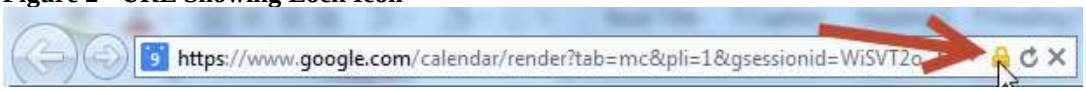
Record the IP address, hostname, common name, SAN entries, and issuer details for all the certificates you encounter when using your application.

To begin, create a list of unique URLs commonly used in the service. This may require some care and planning as Google Apps (and other SaaS applications) consists of multiple services (for example mail, documents, calendar, APIs, etc.) that are on different servers each involving different SSL certificates.

Note: You do not need the full URL, just the portion to the left of the first "/" (e.g., www.google.com in the example below)

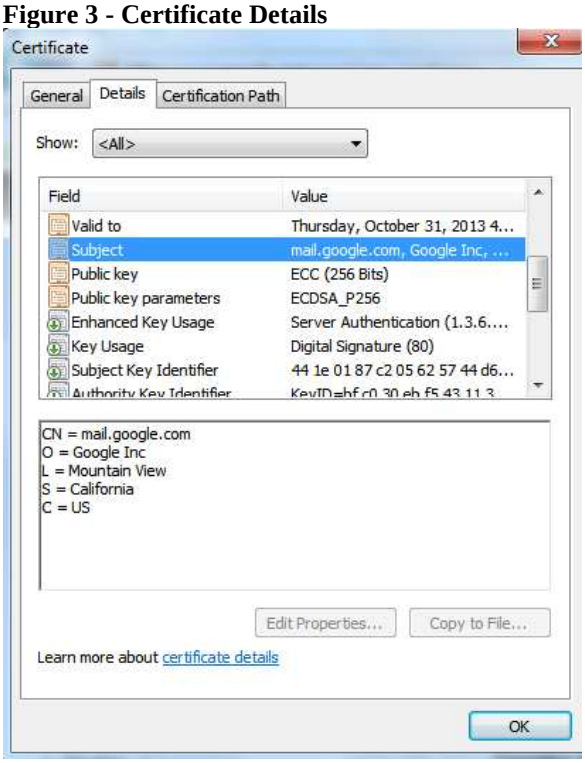
In Internet Explorer, you can browse to the secured site and click on the lock icon as shown.

Figure 2 - URL Showing Lock Icon



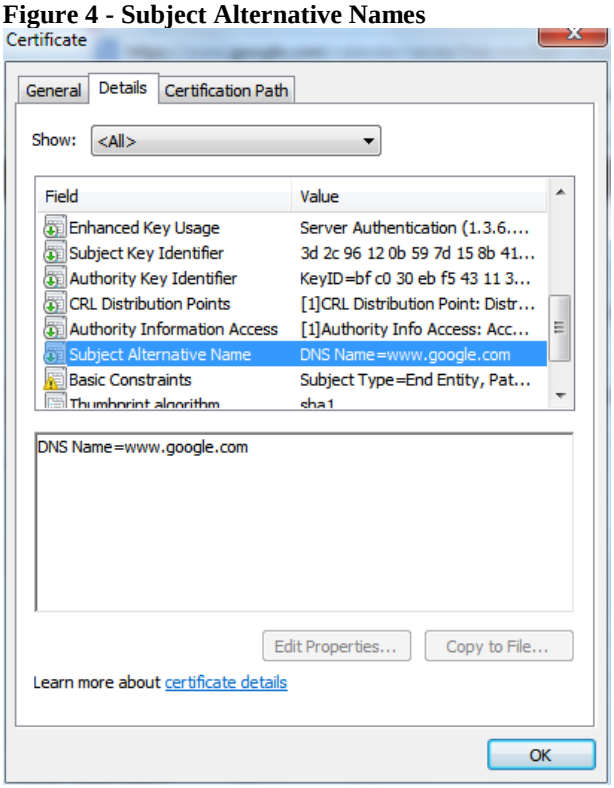
On the pop-up window that appears, select View Certificates and you will see a windows showing details on of the certificate. Click on the *Details* tab, and then *Subject* to see the Common name and other information provided by the issuer.

Figure 3 - Certificate Details



Scroll down the list to see the *Subject Alterative Names* for this certificate as shown:

Figure 4 - Subject Alternative Names

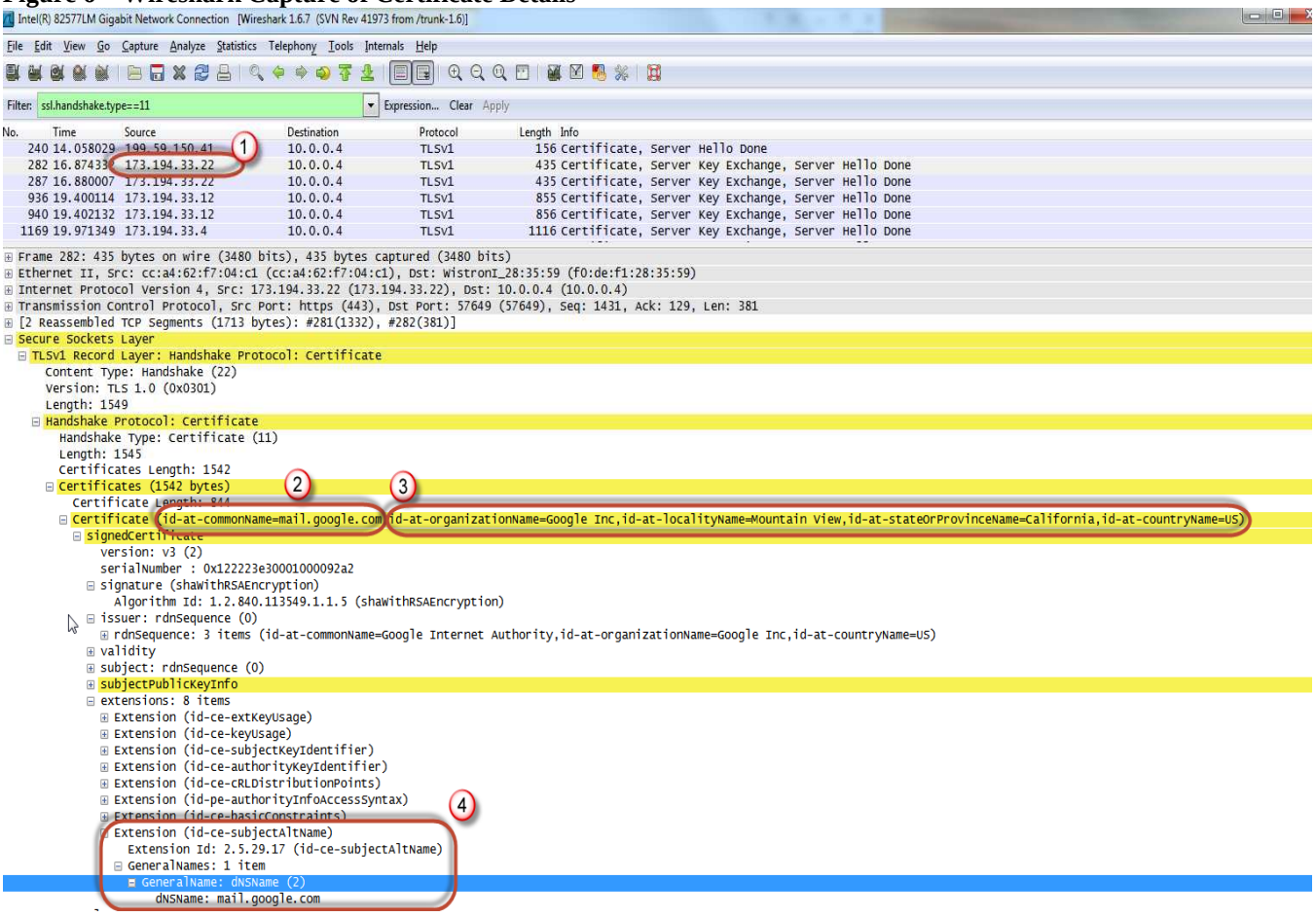


While it is easier to view the details of the certificate in browser, you should not rely on the browser to discover all the certificates that could be in use. Some pages may have links to other SSL connections that are not apparent from the top level URL. The most reliable way to discover all the certificates is to use Wireshark to capture the SSL handshake.

To use Wireshark to capture the SSL handshake:

1. Ensure that all applications are closed before capturing the SSL traffic, because this will ensure that any pre-existing SSL sessions are properly torn down and will correctly re-establish.
2. Launch Wireshark and start capturing the traffic on the relevant interface.
3. Launch Google Apps and perform the same steps a user would perform.
4. Stop capturing on Wireshark and apply the filter ssl.handshake.type == 11

Figure 6 – Wireshark Capture of Certificate Details



Record the following information from each certificate:

- **The IP address:** IP addresses located in the *Source* column are important for use with in-path rules. These addresses may be needed on the client-side Steelhead appliance to optimize the SSL traffic.

  - **Note:** Often you can group the IP addresses together and it may not be necessary to create individual rules. In Figure 6, you can see IP addresses involved in the handshake from the 173.194.33.0/24 subnet.

- **Common name**: The common name is the most important item. You must accurately record the common name for all certificates in use with Google Apps. In Figure 6, the common name is mail.google.com
- **Issuer information**: Issuer information is in the certificate. All certificates have fields used to identify the owner of the certificate such as State and Organization. While these fields are present in a valid certificate, they are not all required to contain information. As a result you will see variations in their use. As a best practice, replicate the details found in the

service certificates in your proxy certificates. This insures you will not have an issue in the unlikely event that application you use is expecting content in these fields.

- **Subject Alterative Names (SAN)**: SAN entries are an important part of the configuration so be sure to gather these details. SAN entries are shown as DNSname in the certificate. In Figure 6, by the item number 4, the only SAN entry(mail.google.com) is the same as the Common Name, so the SAN entry on your proxy certificate is optional. In other cases, you will find multiple entries in the SAN list.

Next, review the common names and SAN entries to see if you can some reduce some of the entries by effective use of wildcards. For example, mail.google.com and www.google.com can be mapped to *.google.com.

Note: A hostname such as *upload.video.google.com* is not matched by *.google.com. Instead, use *.video.google.com to match

The following table shows an example of information you need for creating your proxy certificates.

Host IP	Common Name	SAN Entries	Issuer
173.194.33.3	Google.com	*.google.com *.android.com *.appengine.google.com	CN = Google Internet Authority O = Google Inc C = US

Chapter 3 Creating Proxy Certificates for the Steelhead Appliances

This chapter describes the process and procedures for preparing the Microsoft CA. It includes the following sections:

- Prerequisites
- Process
- Create and Publish a Steelhead Appliance Certificate Template
- Creating a Certificate Signing Request (CSR)
- Generating and Importing a Certificate from the CSR
- Exporting the Certificate with the Private Keys

Prerequisites

- Windows® Server 2008 R2 installed in the role of an Enterprise CA. The process for Windows Server 2012 is similar but not detailed in this paper.
- CA administrator or CA manager permissions. These are provided with the domain administrator group membership.
- Details for the certificates your require as instructed in Chapter 2

Process

Before you can issue certificates containing the CA private key, a new certificate template must be created and made available.

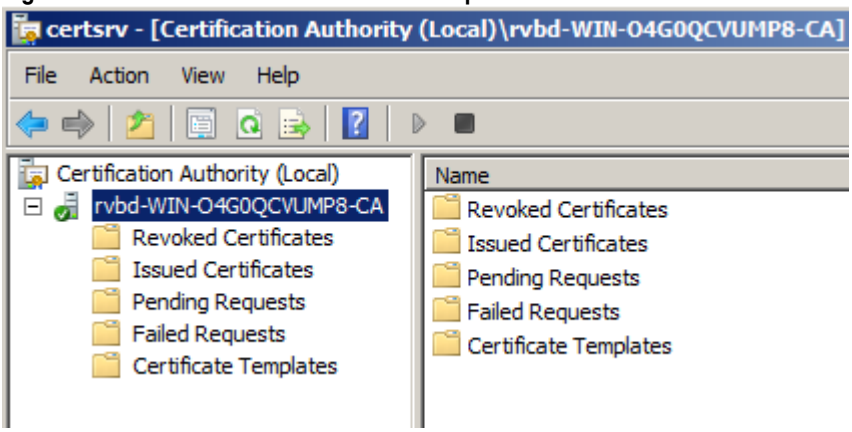
Component	Procedure	Description
Windows Server	Login to a server in the certificate server role	Login using domain administrative credentials.
Various	Create and publish a Steelhead appliance certificate template	Create a new Steelhead appliance certificate template and mark it for publishing. See the section Creating and Publishing a Steelhead Appliance Certificate Template for details
Digicert	Creating a Certificate using Digicert	Use the Digicert utility to create a CSR. Then sign the request using Certreq.exe and import the certificate. Export the certificate with the private keys to complete the process. See the section Creating a Certificate using Digicert for complete instructions.

Creating and Publishing a Steelhead Appliance Certificate Template

This section describes the steps necessary to create a Steelhead appliance certificate template as described in the Process section above.

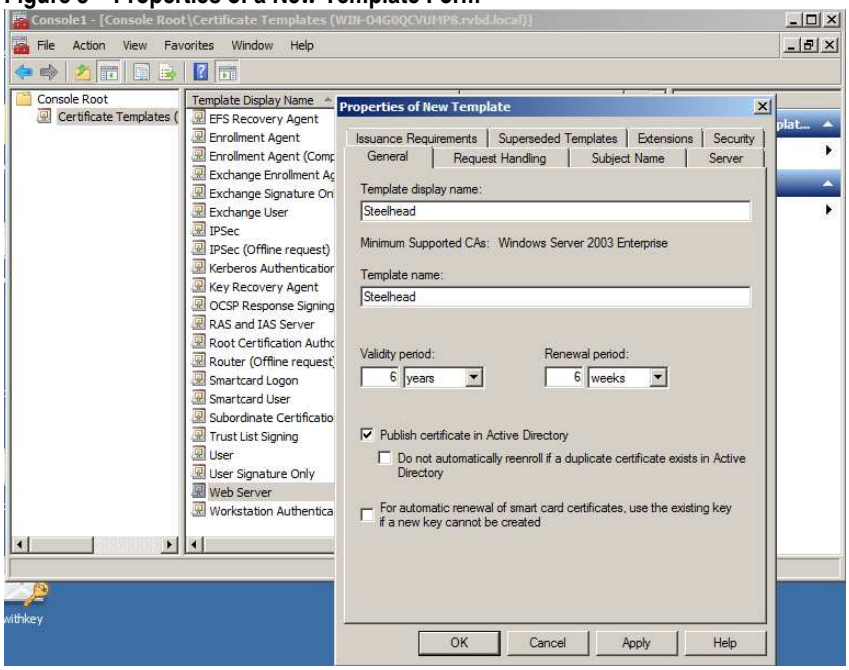
1. Click *Start>Administrative Tools> Certification Authority*. If this option is not available, stop and login to a server hosting the CA role.
2. On the left side of the MMC, expand the node showing the server name as shown in Figure 7

Figure 7 – certsrv MMC with Server Name Expanded



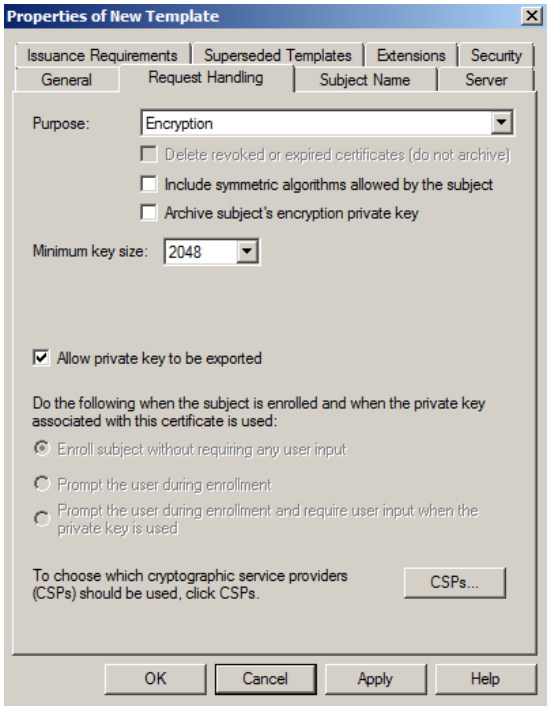
3. Right-click the *Certificate Templates* node and select **Manage.** You will see a list of certificate templates,
4. Right-click the **Web Server Template** and select **Duplicate Template.**
5. Select **Windows Server 2003 Enterprise** and click **OK**. You will see a screen similar to the one shown in Figure 8.

Figure 8 – Properties of a New Template Form



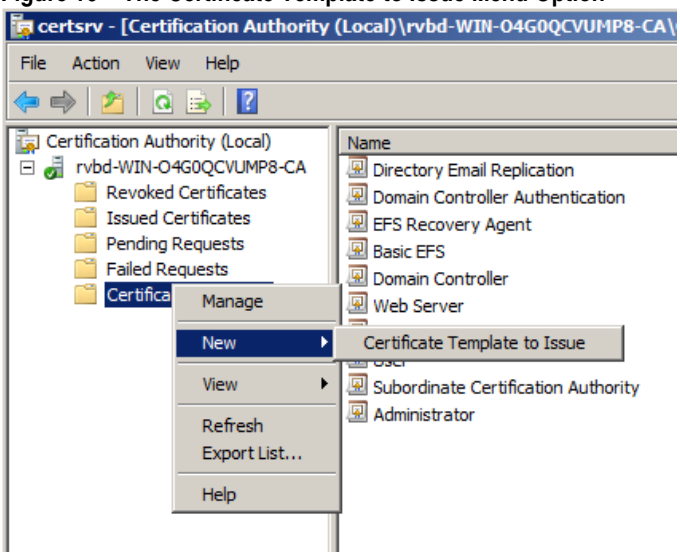
6. For *Template display name*, enter Steelhead. Take special note of the validity period. The default is 2 years. You must renew and reinstall the certificate on the Steelhead appliance at the end of this period. Consider extending the period to reduce administrative overhead.
7. Check the box labeled *Publish certificate in Active Directory*.
8. Select the **Request Handling** tab. You will see a screen similar to Figure 9

Figure 9 – Request Handling Tab



9. Click on the pull-down list next to *Purpose* and select **Encryption.**
10. Check the box labeled *Allow private key to be exported.*
11. Click **Extensions.**
12. Click **Add.**
13. Select **Client Authentication** then click **OK** and **OK** again.
14. Make any other changes you need to comply with your company policy, then click **OK.**
15. Close the Certificate Templates list window.
16. In the CA MMC, right click on **Certificate Templates** and select **New> Certificate Template** to issue.
17. Select *Steelhead* from the list of templates and click **OK**.

Figure 10 – The Certificate Template to Issue Menu Option



Note: There may be some delay between creating the certificate template and the time it appears in the list depending on your Active Directory conditions.

18. Close all open windows.

You have now created a certificate template named *Steelhead* that can be used to issue proxy and peering certificates for your Steelhead appliances.

Creating a Certificate Using the DigiCert Utility

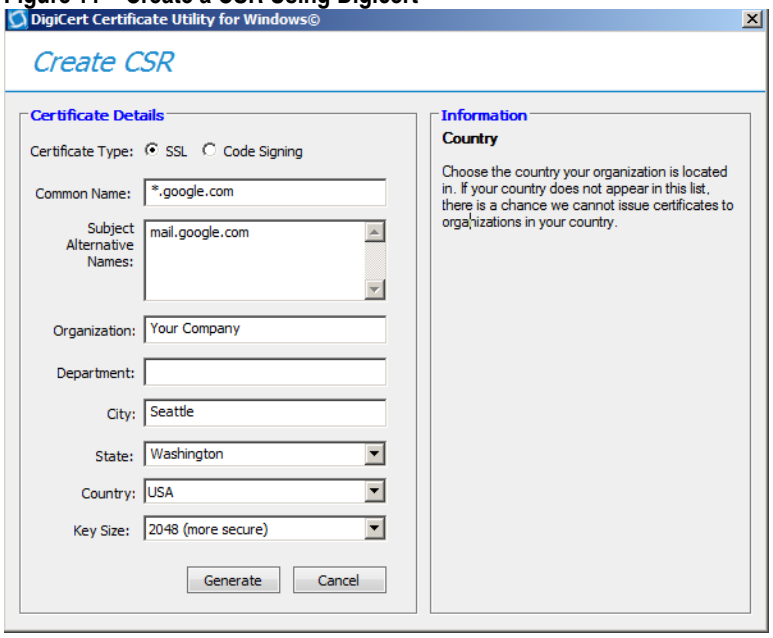
Using the data gathered in Chapter 2, you can now create a CSR and sign it using the Steelhead appliance certificate template. In the end, you have certificate with the private keys of the CA that can be imported into the server-side Steelhead appliance.

Note: The CSR can be created from any computer.

There are numerous ways to create a CSR. Microsoft provides utilities such as the MMC and CertReq.exe commands for this purpose. A free utility from https://www.digicert.com/util/ makes the entire process much. Details on how to use the Microsoft methodologies are located in Appendix A

1. Download and install the Digicert utility.
2. Launch Digicert and select **Create CSR**.
3. Complete the certificate details using the information gathered in Chapter 2.

Figure 11 – Create a CSR Using Digicert



4. Click **Generate**.
5. Click **Save to File** and save the CSR to a file such as Steelhead.txt

Signing the Certificate

Next, use the Microsoft Certreq utility to sign the certificate request. Open an administrative command prompt then use Certreq and specify the CSR and Steelhead certificate template created earlier as follows:

```
Certreq –submit –attrib “CertificateTemplate:Steelhead” <CSRfilename>
<Certificate Filename>
```

Note: If you receive an error stating the certificate was not issued because of the server policy, make sure that you are logged on as a user with permissions to issue certificates. A domain administrator will usually have the necessary permissions.

Import the Certificate

1. From the main Digicert menu, select **Import.**
2. Browse the file or provide the filename and click **Next.**
3. Enter a *Friendly Name* when prompted. This is used to help identify the certificate in the user interface and click **Enter.**

Note: You will see a message stating that you need to go to IIS to assign the certificate. This can be ignored.

Exporting the Certificate with the Private Keys

When you imported the certificate to the CA, it was associated with the private keys of the CA. The Steelhead appliance certificate template allows you to export this certificate with private keys. This can be easily be done in DigiCert as follows:

1. In DigiCert, select the certificate, then click the **Export Certificate** button. Make sure that “Yes, export the private key” is checked.
2. Click **Next** and enter a password to associate with the certificate.

Note: Be certain you remember this password as it cannot be recovered

3. Specify the location of the file.

You now have a certificate that is ready for importing into the server-side Steelhead appliance.

For instructions on importing the certificate, see the Steelhead Deployment Guide – Protocols chapter on SSL deployment.

Appendix A
Using Microsoft Tools to Create a Proxy Certificate

The main body of the paper describes how to use DigiCert to create a CSR, import the certificate, and expert the certificate with private keys. Some organizations may not permit the use of tools such a DigiCert to work with certificates. Microsoft provides utilities that allow you to achieve the same results. This appendix details how to create a CSR, sign the certificate request, import the certificate, and export the certificate with private keys using just the tools available in Windows Server 2008 R2.

This appendix includes the following sections:

- Creating a CSR
- Sign, import, then export the certificate

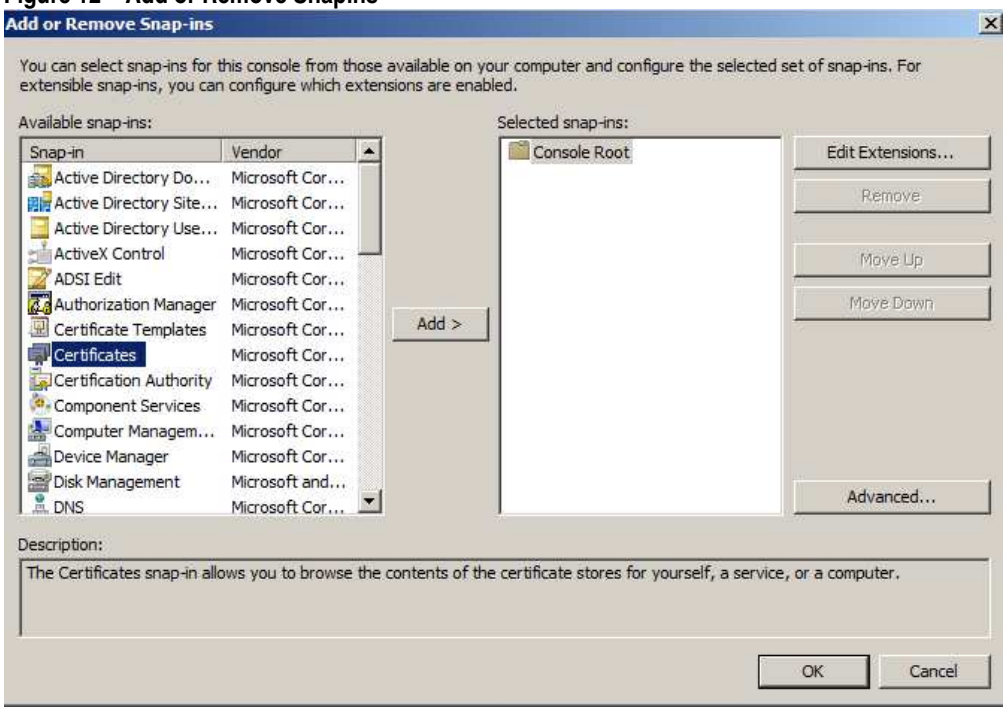
Creating a CSR

You can create a CSR using either the Microsoft Management Console for certificates or by using the Certreq.exe command. This section covers both options. The end result is a CSR file that needs to be signed by the CA.

Using the Microsoft MMC to Create a CSR

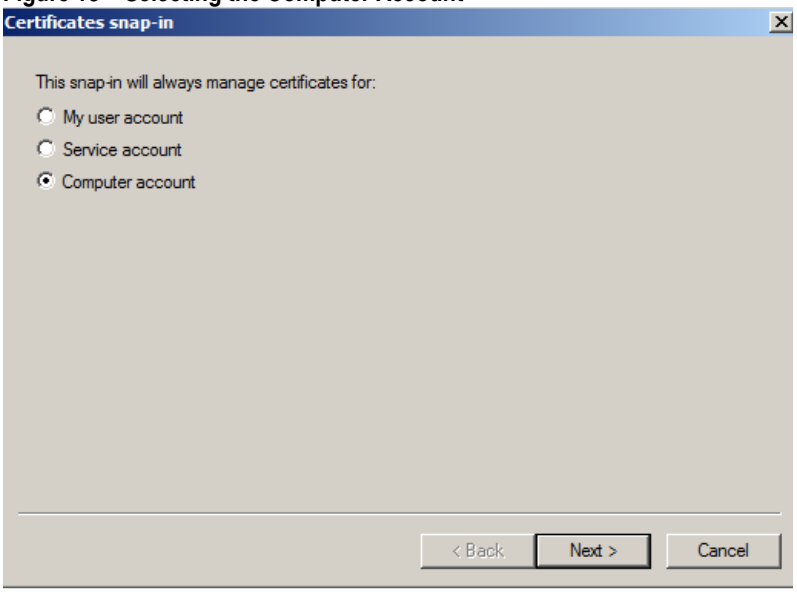
1. Click **Start**. In the Run text box, type *MMC* and press enter.
2. From the MMC main menu, click File> Add Remove Snap In.
3. From the list of available snap-ins, select **Certificates** and click **Add**.

Figure 12 – Add or Remove Snapins



When you click Add, you will see a prompt as shown in Figure 13.

Figure 13 – Selecting the Computer Account

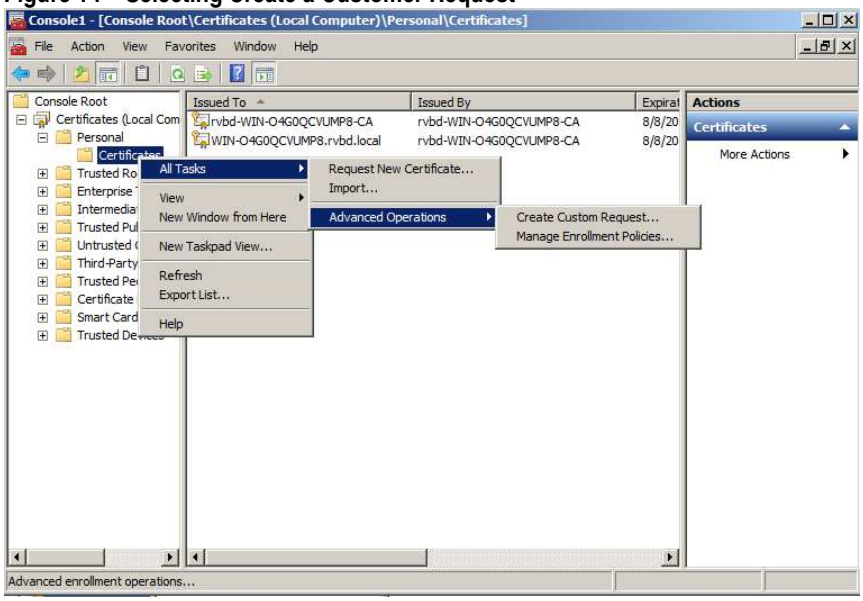


4. Select *Computer account* and click **Next.**

Note: The default selection, *My user account* will not work for this process

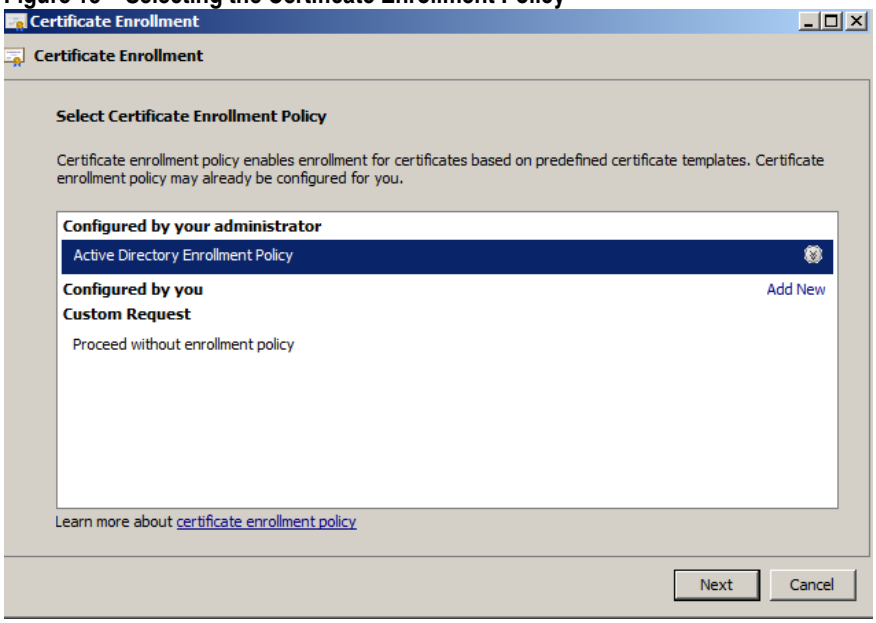
5. On the Select Computer window (not shown) accept the default *Local computer* then click **Finish.** Click **OK.**
6. Expand **Certificates>Personal** then click the **Certificates** folder.
7. Right click on the **Certificates** folder and select **Advanced Operations** then **Create Custom Request** as shown in Figure 14.

Figure 14 – Selecting Create a Customer Request



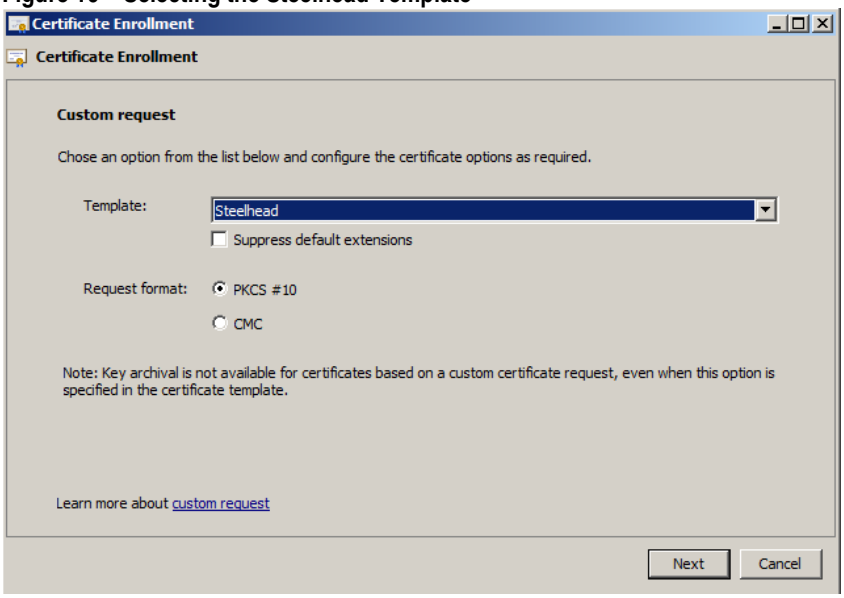
8. Click **Next**. On *the Select Certificate Enrollment Policy* screen **select Active Directory Enrollment Policy** as shown in Figure 15 and click **Next**.

Figure 15 – Selecting the Certificate Enrollment Policy



9. On the *Custom Request* page, under *Templates* select **Steelhead**.

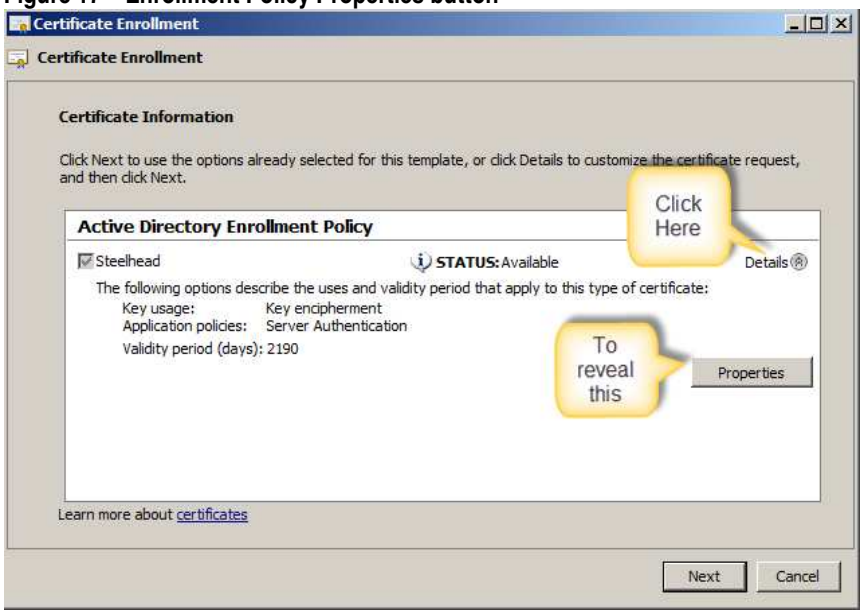
Figure 16 – Selecting the Steelhead Template



10. Make sure *PKCS#10* is selected for the request format and click **Next**.

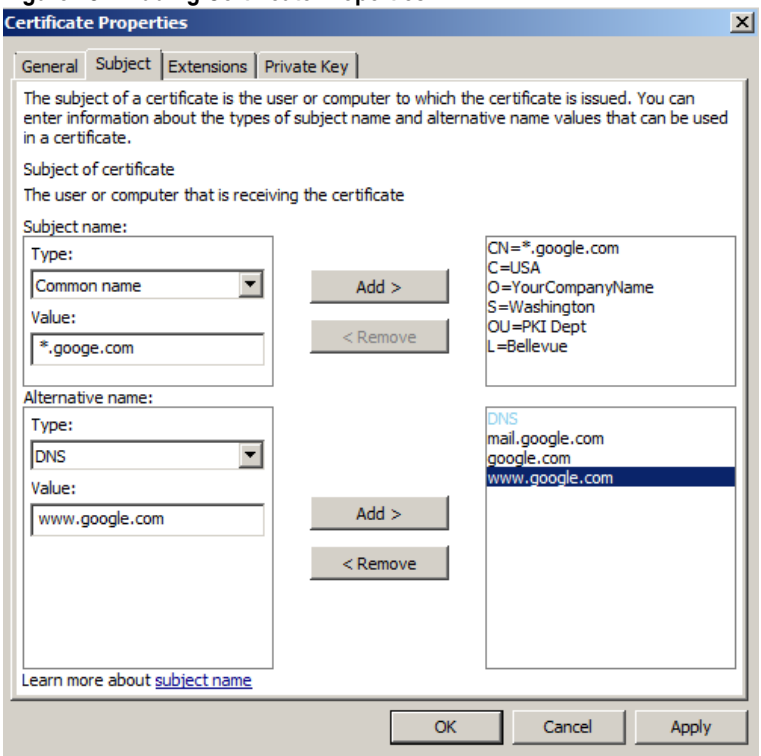
11. The default appearance of the *Certificate Information* window shown in Figure 17 shows no options. Click on the small **Details** label next to the Click Here balloon in Figure 17. You will then see a *Properties* button which opens up new options.

Figure 17 – Enrollment Policy Properties button



12. Click on the **Properties** button.
13. Click on the **Subject** tab. You can fill in details for the certificate by clicking on **Type** and selecting a field, then entering the value for the selected field in the *Value* textbox. Click **Add** to move the entry to the box appearing on the right.

Figure 18 – Adding Certificate Properties



Note: SAN entries are created by adding the DNS type under the Alterative name box.

14. When you have completed you form, click **OK.** No other entries are needed because you specified the Steelhead appliance certificate template that adds needed properties for you.

15. Enter a filename and designate the location to store the CSR, then click **Finish**.

16. Close the MMC. When prompted, provide a name for this MMC such as LocalCerts, as you will use it later.

Using the Certreq.exe Command to Create a CRS

Microsoft provides Certreq.exe on server operating systems to quickly create a .REQ file. The REQ file is a CSR file that can be signed by your Enterprise CA or submitted to well-known CA such as VeriSign.

1. Login to the server acting as your enterprise CA with administrator credentials.
2. Create a working folder and then open an administrator command prompt to work in that folder.
3. Open Notepad and enter content similar to the following substituting your information in the NewRequest and Extensions section.

```
[NewRequest]
Subject="CN=*.google.com,O=MyCompanyName,OU=PKIDept,L=Bellevue,S=Washing
ton,C=US"
[Extensions]
2.5.29.17 = "{text}"
_continue_= "dns=mail.google.com&"
_continue_ = "dns=www.google.com&"
[RequestAttributes]
CertificateTemplate=”Steelhead”
```

4. The [Extension] section allows for SAN entries to be added to the certificate. Add additional _continue_ entries as needed.
5. Save the file as *newcert.inf.*
6. At the command prompt type `certreq –new newcert.inf newcert.req` and press enter. This command is not case sensitive. The newcert.req file will appear in your working folder.

You can find details on the INF file structure at http://technet.microsoft.com/library/cc725793.aspx.

Sign, Import, then Export the Certificate

In this step, you generate a sign the certificate from the CSR and import the certificate to the personal certificate store of the CA. In this way, the certificate is associated with the CA private key.

Signing the Certificate

From an administrative command prompt, type:

```
Certreq –submit –attrib <CSRfilename> <Certificate Filename>
```

In this example, the CSRFilename would be newcert.req from the previous steps. After entering this command you will have .CER file in your working folder. You can double click this file and review the contents in the UI.

Importing the Certificate

The act of importing the certificate associates the server private keys with the certificate.

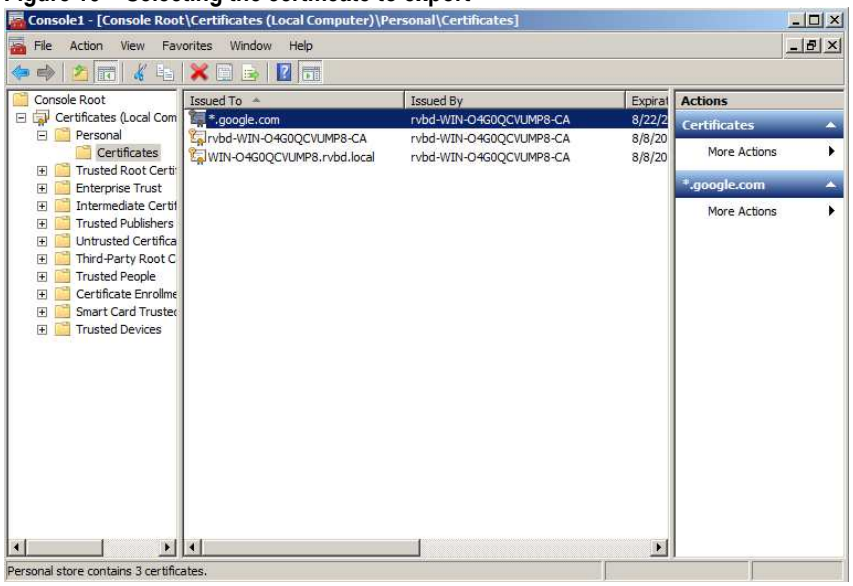
At an administrative command prompt, type:

```
Certreq –accept <Certificate Filename>
```

Exporting the Certificate with the Private Keys

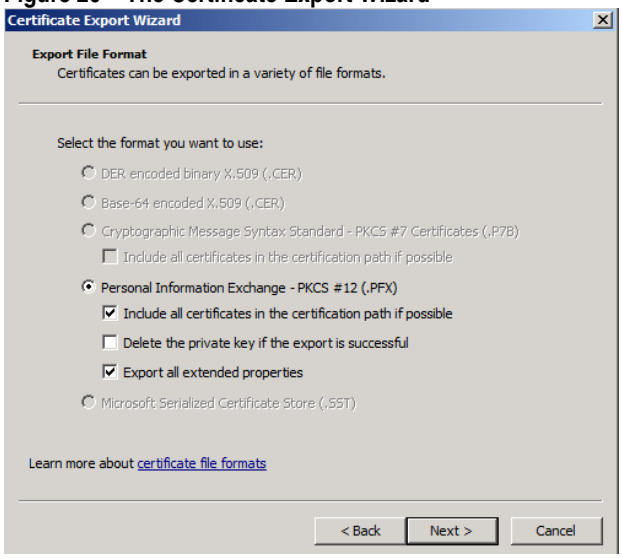
1. Click Start and enter the name of the MMC you created as detailed in Creating a Certificate Signing Request.
2. If necessary, expand *Certificates* then expand Personal then click the **Certificates** folder.

Figure 19 – Selecting the certificate to export



3. Double click on the certificate.
4. Click **Details** then click **Copy to File.**
5. The *Certificate Export Wizard* launches. Click **Next.**
6. On the **Export a Private Key** screen, select *Yes, export the private key* and click **Next.**
7. On the **Export File Format** screen, you may need to select *Include all certificates in the certificate path if possible* if a subordinate CA issued your certificate. Also, *Export all extended properties* may be needed for additional validation or security checks if required by your organization.

Figure 20 – The Certificate Export Wizard



8. On the *Password* screen, enter a password used to unlock the file. The exported file cannot be imported without password.

Note: Be certain you remember this password as it cannot be recovered

9. On the *File to Export* screen, browse to the location to place the file and type a file name. The file extension will be .pfx
10. Click **Finish** and then **OK**.

You now have a proxy certificate you can import into the server-side Steelhead appliance.

For instructions on importing the certificate see the Steelhead Deployment Guide – Protocols chapter on SSL deployment.

riverbed
Think fast.

Riverbed Technology, Inc.
199 Fremont Street
San Francisco, CA 94105
Tel: (415) 247-8800
www.riverbed.com

Riverbed Technology Ltd.
One Thames Valley
Wokingham Road, Level 2
Bracknell. RG42 1NG
United Kingdom
Tel: +44 1344 31 7100

Riverbed Technology Pte. Ltd.
391A Orchard Road #22-06/10
Ngee Ann City Tower A
Singapore 238873
Tel: +65 6508-7400

Riverbed Technology K.K.
Shiba-Koen Plaza Building 9F
3-6-9, Shiba, Minato-ku
Tokyo, Japan 105-0014
Tel: +81 3 5419 1990