

Symantec™ PKI Client

Writing Post-processing Scripts Guide

V2.10

Symantec™ PKI Client Writing Post-processing Scripts Guide

Legal Notice

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Last updated [November 21, 2013](#)

Legal Notice

Copyright © 2011 - 2013 Symantec Corporation. All rights reserved

Symantec, the Symantec Logo, and the Checkmark Logo are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. VeriSign, VeriSign Trust, and other related marks are the trademarks or registered trademarks of VeriSign, Inc. or its affiliates or subsidiaries in the U.S. and other countries and licensed to Symantec Corporation. Other names may be trademarks of their respective owners. The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Symantec Corporation and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. SYMANTEC CORPORATION SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation", as applicable, and any successor regulations. Any use, modification, reproduction, release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement. This document may describe features and/or functionality not present in your software or your service agreement. Contact your account representative to learn more about what is available with this Symantec® product.

Symantec Corporation

350 Ellis Street Mountain View, CA 94043

<http://www.symantec.com>

<http://www.symauth.com/support/contact/index.html#support4>

Chapter 1	Writing Post Processing Scripts	1
	BAT File Format	1
	SH File Format	1
	Script File Parameters	2
	Special Considerations	2
	Sample BAT Files	2
	Sample BAT File Executions	3
	Sample SH File	4
	Sample SH File Executions	5
	Modifying Template Files for Specific Implementations	5
	Template File Configuration Values	6
Appendix A	Certificate Properties Utility	7
	Certificate Properties Utility Usage (Windows)	7
	Certificate Properties Commands for Batch	8
	Sample Certificate Properties Utility Usage	9
	Certificate Properties Utility Usage (Mac)	9
	Certificate Properties Commands for Shell	10
	Sample Certificate Properties Utility Usage	11
Appendix B	Error Codes and Troubleshooting	13
	Script File Error Codes	13
Index	17

Writing Post Processing Scripts

Once the PKI Client has completed operations on a certificate, your application will need to be able to consume them. You can write BAT (batch) for Windows or SH (shell) script files for Mac to notify your applications of the certificate status and locations, and that the certificates are available for use.

Prior to calling the script file, the PKI Client writes certificate data to files which will be made accessible to the script files. The script files will use this data to perform the requested operation. The fully-qualified path to the certificate files will be provided to the script file. The certificate files will be deleted after post-processing is completed.

This guide describes how to write script files to notify your applications of certificate status, location, and availability after the following certificate operations:

- Enroll (new certificate)
- Import (existing certificate)
- Renew (new certificate, old certificate)
- Smart Card Insert (certificate on security device)
- Diagnostic Mode

Note: PKI Client allows users to import expired certificates (this is sometimes useful, for example, to allow users to access email previously encrypted by an expired certificate). To verify the validity of a certificate before importing it, use the Certificate Property utility. Refer to Appendix A "Certificate Properties Utility," for information about this utility.

Once your script files are prepared, you upload them to PKI Manager and assign them to one or more certificate profiles from the *Manage Profiles* page (you can also remove them from a specific certificate profile on the *Manage this profile* page for that certificate profile). Once assigned, the script files will be run for every certificate issued from that profile.

Refer to PKI Manager and its associated help for procedures on uploading the script files and assigning them to certificate profiles.

BAT File Format

Windows BAT files must end with a blank line and begin with the following line:

```
@ECHO OFF
```

SH File Format

Mac SH files must end with a blank line and begin with the appropriate hash-bang (#!), usually:

```
#!/bin/sh
```

Additionally, all error codes must be between 1 and 255. The code 195 is reserved and should not be used.

Script File Parameters

The PKI client passes the following information to your script file:

- The operation, as mentioned in the operations section.
- The path to the PKI Client install directory (necessary if the script file needs to locate other dependencies, such as binaries).
- The path to the new or imported certificate file, as appropriate.
- The path to the old certificate file (for renewal only; otherwise this parameter will not be present).
- The path to the root certificate of the issuing CA.
- The paths to any intermediate certificates between the root and the user certificate.

Special Considerations

When creating your script files, refer to the following special considerations:

- If your script file depends upon or calls other applications, these applications must be present in the <PKI_Client_install_dir>\BERETTA\BIN directory.
- Your script file can invoke a command line utility obtain information about a specific certificate. Refer to Appendix A "Certificate Properties Utility," for more information on this utility.
- Managed PKI does not control the order in which script files are run. If you use multiple script files to perform operations on a single certificate, all of the script files must be written such that they do not rely on the output of other script files, nor cause any side-effects that will affect the other script files.

If you want to control the order of certificate operations, you will need to merge all certificate operations into a single script file and use that script file to control the dependencies between operations.

Sample BAT Files

The following sample BAT file writes example output to the screen, although will not actually install a certificate.

The PKI Client will execute your script file and will pass the following parameters to your script file, in this order:

- Operation (Enroll, Import or Renew)
- PKI Client installation directory
- The user's certificate that was just enrolled, imported, or renewed
- The user's expiring certificate (only applies to Renew operations)
- The root CA certificate
- The intermediate CA certificates

Note that for successful operations you must end with an exit code of 0, and for any error conditions, you must end with an exit code of 1000 or higher.

Non-zero exit codes will display to your end users, as well as be written to the beretta.log file. Use a different error code for each type of error to aid in troubleshooting issues.

```
@ECHO OFF
```

```
if "%1" == "Enroll" (GOTO enroll)
if "%1" == "Import" (GOTO import)
if "%1" == "Renew" (GOTO renew)
```

```
ECHO Operation not supported: %1
```

```
EXIT /B 1

:enroll
ECHO Adding new certificate
ECHO The PKI Directory is %2
ECHO The New Certificate is %3
ECHO The Root CA Certificate is %4
ECHO The Intermediate Certificates are %5, %6, %7, %8, %9
EXIT /B 0

:import
ECHO Adding imported certificate
ECHO The PKI Directory is %2
ECHO The Imported Certificate is %3
ECHO The Root CA Certificate is %4
ECHO The Intermediate Certificates are %5, %6, %7, %8, %9
EXIT /B 0

:renew
ECHO Renewing certificate
ECHO The PKI Directory is %2
ECHO The New Certificate is %3
ECHO The Old Certificate is %4
ECHO The Root CA Certificate is %5
ECHO The Intermediate Certificates are %6, %7, %8, %9
EXIT /B 0
```

Sample BAT File Executions

```
*Enroll:
Sample.bat Enroll "C:\Program Files\Symantec\PKI Client\" "C:\Somepath\New.cer"
"C:\Somepath\Root.cer" "C:\Somepath\Intermediate1.cer" "C:\Somepath\Intermediate2.cer"
Output:
Adding new certificate
The PKI Client Directory is "C:\Program Files\Symantec\PKI Client\"
The New Certificate is "C:\Somepath\New.cer"
The Root CA Certificate is "C:\Somepath\Root.cer"
The Intermediate Certificates are "C:\Somepath\Intermediate1.cer",
"C:\Somepath\Intermediate2.cer", , ,

*Import:
Sample.bat Import "C:\Program Files\Symantec\PKI Client\" "C:\Somepath\Import.cer"
"C:\Somepath\Root.cer" "C:\Somepath\Intermediate1.cer" "C:\Somepath\Intermediate2.cer"
Output:
Adding imported certificate
The PKI Client Directory is "C:\Program Files\Symantec\PKI Client\"
The Imported Certificate is "C:\Somepath\Import.cer"
The Root CA Certificate is "C:\Somepath\Root.cer"
The Intermediate Certificates are "C:\Somepath\Intermediate1.cer",
"C:\Somepath\Intermediate2.cer", , ,

*Renew:
```

```
Sample.bat Renew "C:\Program Files\Symantec\PKI Client\" "C:\Somepath\New.cer"
"C:\Somepath\Old.cer" "C:\Somepath\Root.cer" "C:\Somepath\Intermediate1.cer"
"C:\Somepath\Intermediate2.cer"
Output:
Renewing certificate
The PKI Client Directory is "C:\Program Files\Symantec\PKI Client\"
The New Certificate is "C:\Somepath\New.cer"
The New Certificate is "C:\Somepath\Old.cer"
The Root CA Certificate is "C:\Somepath\Root.cer"
The Intermediate Certificates are "C:\Somepath\Intermediate1.cer",
"C:\Somepath\Intermediate2.cer", ,

*Other:
Sample.bat Other "C:\Program Files\Symantec\PKI Client\" "C:\Somepath\New.cer"
"C:\Somepath\Root.cer" "C:\Somepath\Intermediate1.cer" "C:\Somepath\Intermediate2.cer"
Output:
Operation not supported: Other
```

Sample SH File

The following sample SH file writes example output to the screen, although will not actually install a certificate.

The PKI Client will execute your script file and will pass the following parameters to your script file, in this order:

- Operation (Enroll, Import or Renew)
- PKI Client installation directory
- The user's certificate that was just enrolled, imported, or renewed
- The user's expiring certificate (only applies to Renew operations)
- The root CA certificate
- The intermediate CA certificates

Note that for successful operations you must end with an exit code of 0, and for any error conditions, you must end with an exit code of 1000 or higher.

Non-zero exit codes will display to your end users, as well as be written to the beretta.log file. Use a different error code for each type of error to aid in troubleshooting issues.

```
#!/bin/sh

if [ "$1" == "Enroll" ]
    echo Adding new certificate
    echo The PKI Directory is $2
    echo The New Certificate is $3
    echo The Root CA Certificate is $4
    echo The Intermediate Certificates are $5, $6, $7, $8, $9

elif [ "$1" == "Import" ]
    echo Adding imported certificate
    echo The PKI Directory is $2
    echo The Imported Certificate is $3
    echo The Root CA Certificate is $4
    echo The Intermediate Certificates are $5, $6, $7, $8, $9
    exit 0
```

```
elif [ "$1" == "Renew" ]
    echo Renewing certificate
    echo The PKI Directory is $2
    echo The New Certificate is $3
    echo The Old Certificate is $4
    echo The Root CA Certificate is $5
    echo The Intermediate Certificates are $5, $6, $7, $8
    exit 0

else
    echo Operation not supported
    exit 1
fi

exit 0
```

Sample SH File Executions

```
S*Enroll:
./Sample.sh Enroll "/usr/local/lib/tblive-4" "/Somepath/New.cer" "/Somepath/Root.cer" "/
Somepath/Intermediate1.cer" "/Somepath/Intermediate2.cer"
Output:
Adding new certificate
The PKI Client Directory is "/usr/local/lib/tblive-4"
The New Certificate is "/Somepath/New.cer"
The Root CA Certificate is "/Somepath/Root.cer"
The Intermediate Certificates are "/Somepath/Intermediate1.cer",
"/Somepath/Intermediate2.cer", , ,
```

Modifying Template Files for Specific Implementations

Symantec provides template versions of script files that you can modify for your specific implementations.

- 1 Download a template script file from the *Create custom scripts* page of PKI Manager.
- 2 Modify the template file to meet your needs. Typically, you should only need to modify entries between the START CONFIG and END CONFIG headers.

Refer to [“Template File Configuration Values”](#) on page 6 for details on the values you can modify in these template files.

- 3 Upload the modified file to your Managed PKI account from the *Create custom script* page.
- 4 Assign the newly uploaded file to the appropriate certificate profile on the *Manage certificate profile* page.

Template File Configuration Values

This section describes the values you can modify in the script template files.

Table 1-1 Templates and configuration values

Template	Usage	Values to Modify
ADPub.bat*	Publish the certificate to an Active Directory. This specific script template file is supported on Windows only for certificates stored at Symantec. (For certificates enrolled using PKI Enterprise Gateway or the Autoenrollment Server, use profile creation).	No configuration is necessary to use this template.
Cisco.bat	Configure a certificate to connect to a Cisco router. Supported on Windows only	<ul style="list-style-type: none"> ■ ProfileName - The name of the Cisco batch file. ■ Host - IP address of the Cisco VPN gateway host.
Juniper.bat and Juniper.sh	Configure a certificate to connect to a Juniper router. Supported on Windows (.bat) or Mac OSX (.sh)	<ul style="list-style-type: none"> ■ Filename - The name of the VPN batch file. ■ Realm - The Juniper VPN authentication realm. ■ URL - URL or IP address of the Juniper VPN gateway. ■ MIN - (true false) Whether to run the BAT file minimized.
Outlook.bat	Configure Outlook's security profile to use the certificate for signing and encrypting. Supported on Windows only	No configuration is necessary to use this template.
WiFi.bat	Configure a certificate to connect to a Wi-Fi network. Supported on Windows only	<ul style="list-style-type: none"> ■ ProfileName - The name of the WiFi profile (usually the same as the SSID, but on Windows XP or later, these can be different). This value is displayed to the user. ■ SSID - The SSID of your wireless network. ■ nonBroadcast - (true false) Whether the network broadcasts the SSID. If set to true, the SSID is not broadcast. ■ connectionMode - (manual auto) Whether the computer should automatically connect to this network when it is in range.

* For certificates enrolled using PKI Enterprise Gateway or the Autoenrollment Server, use profile creation. This functionality is configured under **Advanced certificate options > Publish to company directory**.

Certificate Properties Utility

You can invoke the Certificate Properties utility from within a script file to obtain the following information about a specific certificate:

- Expiration date and expiration status (expired or not)
- Subject and common name
- Issuer and serial number
- Policy object identifier (OID) for the certificate.

Certificate Properties Utility Usage (Windows)

To invoke the Certificate Properties utility:

- 1 Set the PKI directory in your BAT file using the CD command (the second argument, %2, should always be the PKI directory in your BAT file). For example:

```
cd %2
```

This must be done only once in your BAT file unless some other operation uses the cd command. In that case, you must use the cd command to reset the directory.

- 2 Add the following commands to your BAT file:

```
tblive-4-helper-console-x86.exe DSM\cmdlineutil.dsm [operation]  
[argument] [path to certificate file]
```

[Table A-1](#) lists the operations and arguments supported by this command.

Table A-1 Certificate Properties utility options

Operation	Argument	Description
USAGE	None	Displays the utility usage and help file.
CERTPROPERTY	EXPIRED	Returns the expiration status of the certificate: <ul style="list-style-type: none"> TRUE indicates that the certificate is expired. FALSE indicates that the certificate is not expired.
	EXPIRATION_DATE	Returns the certificate expiration date. The date format will be in the default system date format. To change the format of the response, add an argument to the end of the command to identify the date format (use a date format string according to the C function strftime).
	SUBJECT	Returns the subject name of the certificate.
	SUBJECT_COMMON_NAME	Returns the common name in the certificate's subject name.
	ISSUER	Returns the name of the certificate issuer.
	SERIAL_NUMBER	Returns the certificate serial number.
	POLICY_OID	Returns the certificate's policy object identifier (OID). The policy OID should match the OID listed for the associated certificate profile in PKI Manager.

Certificate Properties Commands for Batch

The following are example Certificate Properties commands. A full, contextual usage example is provided in Sample Certificate Properties Utility Usage on page 12.

Note: Due to page limitations, the commands shown here wrap to multiple lines. However, each command is a single line. Separate command in your BAT file should always be on one line.

- Obtaining the expired status of a certificate:

```
FOR /F "tokens=*" %A IN ('tblive-4-helper-console-x86.exe
DSM\cmdlineutil.dsm CERTPROPERTY EXPIRED %3') DO SET Expired=%A
```
- Obtaining the expiration date of a certificate:

```
FOR /F "tokens=*" %A IN ('tblive-4-helper-console-x86.exe
DSM\cmdlineutil.dsm CERTPROPERTY EXPIRATION_DATE %3 "%Y-%m-%d
%H:%M:%S"') DO SET ExpirationDate=%A
```

Note: You must use double-percent characters when running strftime commands in a BAT file. Also, avoid using ^ and & characters in date format strings.

- Obtaining the subject name of a certificate:

```
FOR /F "tokens=*" %A IN ('tblive-4-helper-console-x86.exe
DSM\cmdlineutil.dsm CERTPROPERTY SUBJECT %3') DO SET Subject=%A
```
- Obtaining the common name of a certificate:

```
FOR /F "tokens=*" %A IN ('tblive-4-helper-console-x86.exe
DSM\cmdlineutil.dsm CERTPROPERTY SUBJECT_COMMON_NAME %3') DO SET
CertCommonName=%A
```
- Obtaining the issuer of a certificate.

```
FOR /F "tokens=*" %%A IN ('tblive-4-helper-console-x86.exe DSM\cmdlineutil.dsm  
CERTPROPERTY ISSUER %3') DO SET Issuer=%%A
```

- Obtaining the serial number of a new certificate.

```
FOR /F "tokens=*" %%A IN ('tblive-4-helper-console-x86.exe  
DSM\cmdlineutil.dsm CERTPROPERTY SERIAL_NUMBER %3') DO SET  
SerialNumber=%%A
```

- Obtaining the policy OID of a certificate.

```
FOR /F "tokens=*" %%A IN ('tblive-4-helper-console-x86.exe  
DSM\cmdlineutil.dsm CERTPROPERTY POLICY_OID %3') DO SET PolicyOID=%%A
```

Sample Certificate Properties Utility Usage

```
REM Initialize variables  
SET Expired=  
SET ERRORLEVEL=0  
  
REM Change to PKI Client installation directory  
CD %2  
  
REM Call utility to determine expiration status  
FOR /F "tokens=*" %%A IN ('tblive-4-helper-console-x86.exe ^  
DSM\cmdlineutil.dsm CERTPROPERTY EXPIRED %3') DO SET ^ Expired=%%A  
  
REM Check for errors from the utility  
IF %ERRORLEVEL% GTR 0 EXIT /B %ERRORLEVEL%  
  
REM Check for an empty result from the utility  
IF "%Expired%"==" " EXIT /B 1000  
  
REM Check the result, and quit if certificate is expired  
IF "%Expired%"=="TRUE" EXIT /B 0  
  
REM Ready for further processing.  
ECHO Your certificate is valid.
```

Certificate Properties Utility Usage (Mac)

To invoke the Certificate Properties utility, add the following commands to your SH file:

```
"$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm [operation] [argument] [path to certificate  
file]
```

[Table A-2](#) lists the operations and arguments supported by this command.

Table A-2 Certificate Properties utility options (Mac)

Operation	Argument	Description
USAGE	None	Displays the utility usage and help file.
CERTPROPERTY	EXPIRED	Returns the expiration status of the certificate: <ul style="list-style-type: none">■ TRUE indicates that the certificate is expired.■ FALSE indicates that the certificate is not expired.
	EXPIRATION_DATE	Returns the certificate expiration date. The date format will be in the default system date format. To change the format of the response, add an argument to the end of the command to identify the date format (use a date format string according to the C function strftime).
	SUBJECT	Returns the subject name of the certificate.
	SUBJECT_COMMON_NAME	Returns the common name in the certificate's subject name.
	ISSUER	Returns the name of the certificate issuer.
	SERIAL_NUMBER	Returns the certificate serial number.
	POLICY_OID	Returns the certificate's policy object identifier (OID). The policy OID should match the OID listed for the associated certificate profile in PKI Manager.

Certificate Properties Commands for Shell

The following are example Certificate Properties commands. A full, contextual usage example is provided in Sample Certificate Properties Utility Usage on page 12.

Note: Due to page limitations, the commands shown here wrap to multiple lines. However, each command is a single line. Separate command in your SH file should always be on one line.

- Obtaining the expired status of a certificate:

```
Expired="$( "$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY EXPIRED "$3" )"
```
- Obtaining the expiration date of a certificate:

```
ExpirationDate="$( "$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY EXPIRATION_DATE "$3" "%Y-%m-%d %H:%M:%S" )"
```

Note: You must use double-percent characters when running strftime commands in an SH file. Also, avoid using ^ and & characters in date format strings.

- Obtaining the subject name of a certificate:

```
Subject="$( "$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY SUBJECT "$3" )"
```
- Obtaining the common name of a certificate:

```
CertCommonName="$( "$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY SUBJECT_COMMON_NAME "$3" )"
```
- Obtaining the issuer of a certificate.

```
Issuer="$( "$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm CERTPROPERTY ISSUER "$3" )"
```

- Obtaining the serial number of a new certificate.

```
SerialNumber="$("$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY SERIAL_NUMBER "$3")"
```

- Obtaining the policy OID of a certificate.

```
PolicyOID="$("$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm  
CERTPROPERTY POLICY_OID "$3")"
```

Sample Certificate Properties Utility Usage

```
#!/bin/sh

# Call utility to determine expiration status
expired="$("$2"/tblive-4-helper-x86_64 -c cmdlineutil.dsm CERTPROPERTY EXPIRED "$3")"

ErrorCheck=$?

# Check for errors from the utility
if [ $ErrorCheck -ne 0 ] ; then
    exit $ErrorCheck

# Check for an empty result from the utility
elif [ "$expired" == "" ] ; then
    exit 1

# Check the result, and quit if certificate is expired.
elif [ "$expired" == "TRUE" ] ; then
    exit 0
fi

# Ready for further processing
echo Your certificate is valid.
```


Error Codes and Troubleshooting

This chapter describes common issues you may encounter, and provides some solutions.

Script File Error Codes

You may encounter the following errors when processing script files.

Table B-1 Generic script file errors

Error Code	Description
2	Failed to get AES 256 CBC cipher.
3	Failed to initialize AES context.
4	Failed to initialize SHA256 hmac.
5	HMAC hash does not match.
6	Could not open script file. Solution: Verify that the script file is present in the correct location.
7	Script file does not match signature. Solution: The script file has been modified since it was signed.
8	Script file certificate is invalid. Solution: The certificate provided in the script file is corrupted or does not conform to policy for the available signing certificate chain.
9	Script file signing organization not trusted. Solution: The proper organization was not set in the policy. The corresponding log message will contain the expected organization name.
10	Failed to get SHA256 digest.
11	Could not validate script certificate.

Table B-2 Certificate Properties utility errors

Error Code	Description
50	Command line utility could not open certificate file.
51	Invalid certificate property provided to command line utility.
52	Command line utility could not find the specified value.
53	Command line utility could not parse certificate file.
54	Invalid operation for command line utility

The following errors are specific to individual script files. Symantec recommends that you use unique error codes when creating custom scripts. For example, use four-digit error codes starting at 1000.

Table B-3 InstallCA.signed.script file errors

Error Code	Description
100	Could not open certificate file.
101	Could not open Current User's Root certificate store.
102	Could not create certificate context for root certificate.
103	Could not add root certificate to store.
104	Could not open Current User's Intermediate certificate store.
105	Failed to parse intermediate certificate.
106	Could not create certificate context for intermediate certificate.
107	Could not add certificate to store.

Table B-4 RegisterFirefox.signed.script file errors

Error Code	Description
193	Could not rebuild Mozilla Firefox PKCS11 database
194	NSS_DEFAULT_DB_TYPE environment variable not set
196	Could not verify installation of PKCS11 module.
197	Could not locate path to Firefox profiles.
198	Could not read Firefox Install Directory from registry.
199	Could not read Firefox CurrentVersion from registry.
200	Could not add the Certificate Authority Root to Firefox.
201	An error occurred in Firefox's NSS utilities.

Table B-5 ADPub.signed.script file errors

Error Code	Description
300	Failed to parse certificate data.
301	Failure in LDAP Update.

Table B-6 Outlook.signed.script file errors

Error Code	Description
400	Could not open Microsoft Office from registry.
401	Could not open Outlook from registry.
402	Failure in Outlook MAPI.
403	Failed to launch 64-bit process.

Table B-6 Outlook.signed.script file errors (Continued)

Error Code	Description
404	Unable to retrieve a valid bitness value from registry.
405	Failed to parse certificate data.

Table B-7 Juniper.signed.script file errors

Error Code	Description
500	Could not parse certificate
501	Could not find certificate common name.
502	Could not open Network Connect from registry
503	Could not read InstallPath from registry.
504	Could not retrieve Policy OID from certificate
505	Missing authentication realm in configuration file
506	Missing URL in configuration file
507	Missing Filename in configuration file

Table B-8 WiFi.signed.script file errors

Error Code	Description
600	Could not find Profile Name in configuration file
601	Could not find network SSID in configuration file
602	Could not find valid nonBroadcast in configuration file (true/false)
603	Could not find valid connectionMode in configuration file (auto/manual)
604	Invalid wireless profile XML
605	Wireless profile could not be saved

Table B-9 Cisco.signed.script file errors

Error Code	Description
700	Could not retrieve certificate policy OID
701	Could not find certificate common name.
702	Could not find certificate subject
703	Could not find Profile Name in configuration file
704	Could not find Host in configuration file
705	Failed to create Cisco VPN profile

A

- Active Directory template BAT file 6
- ADPub.signed.bat errors 14
- applications 2
- assigning BAT files to certificate profiles 1

B

- BAT file
 - certificate data for 1
 - considerations for 2
 - dependencies 2
 - multiple 2
 - parameters for 2
 - sample 2-3
 - sample execution 3
 - specific implementation 5
 - supporting applications for 2
 - templates 5
- bat file
 - echo off 1
- beretta.log file 2, 4

C

- certificate data for BAT files 1
- certificate file
 - path parameter for 2
 - path to 1
- certificate profiles 1
- Certificate Properties
 - commands 8, 10
 - sample usage 9, 11
- Certificate Properties utility 7
 - errors 13
 - options 8, 10
 - usage of 7, 9
- Cisco template BAT file 6
- Cisco.signed.bat errors 15
- commands for Certificate Profile utility 8, 10
- configuration files for specific implementations 5
- consideration 2

D

- dependencies 2

E

- ECHO OFF 1
- Enroll operation 1
- Enroll operation sample BAT file 2-3
- Enroll operation sample SH file 4-5

- error codes 13-15
 - ADPub.signed.bat 14
 - Certificate Properties utility 13
 - Cisco.signed.bat 15
 - InstallCA.signed.bat 14
 - Juniper.signed.bat 15
 - Outlook.signed.bat 14
 - RegisterFirefox.signed.bat 14
 - WiFi.signed.bat 15
- exit code 2, 4
- expired certificates 1

H

- hash-bang 1

I

- Import operation 1
- Import operation sample BAT file 2-3
- Import operation sample SH file 4-5
- importing expired certificates 1
- install directory parameter 2
- InstallCA.signed.bat errors 14
- intermediate certificate parameter 2
- issuing CA parameter 2

J

- Juniper template BAT file 6
- Juniper template SH file 6
- Juniper.signed.bat errors 15

M

- multiple BAT files 2

O

- operations
 - batch file operations 1
 - order of 2
 - parameter for 2
- order of operations 2
- Outlook template BAT file 6
- Outlook.signed.bat errors 14

P

- parameters in BAT files 2
- path to certificate files 1
- PKI Client
 - install directory 2
 - install directory parameter for 2

PKI Manager 1

R

RegisterFirefox.signed.bat errors 14

Renew operation 1

Renew operation sample BAT file 2-3

Renew operation sample SH file 4-5

root certificate parameter 2

S

sample

BAT file 2-3

BAT file execution 3

Certificate Profile utility usage 9, 11

SH file 4-5

SH file execution 5

SH file

hash-bang 1

sample 4-5

sample execution 5

special considerations 2

supporting applications 2

T

template BAT file 5

Active Directory values for 6

Cisco values for 6

Juniper values for 6

Outlook values for 6

Wi-Fi values for 6

template SH file

Juniper values for 6

U

uploading BAT files to PKI Manager 1

usage for Certificate Properties utility 7, 9

W

Wi-Fi template BAT file 6

WiFi.signed.bat errors 15