

Trust&Go Step by Step Guide -Loading Manifest to AWS-IoT

Table of Contents

1	Introduction	3
1.1	Getting started with Jupyter Notebook Tutorials	3
1	.1.1 Starting Jupyter Notebook	3
1.2	Jupyter Notebook Basics	3
1	.2.1 The Notebook dashboard	3
1.3	Introduction to Jupyter Notebook GUI	4
2	Jupyter Notebook Tutorials	6
3	Manifest Generation Notebook	7
4	Loading Manifest to AWS-IoT	9
4.1	CryptoAuth TrustPlatform Factory reset	. 13

1 Introduction

This document explains step by step process involved in uploading a manifest file to AWS cloud. If you are already familiar with Jupyter Notebook you can skip this section and move to Section 2.

1.1 Getting started with Jupyter Notebook Tutorials

Jupyter Notebook is open source web application which allows you to create documents that contain code that you can execute in place as well as narrative text. It provides GUI elements, ability to execute code in place, ability to add images and gives it the look and feel that normal code files lack.

Jupyter notebooks are mainly used to explain/evaluate code in an interactive way.

1.1.1 Starting Jupyter Notebook

Jupyter notebook can be launched from the Anaconda Navigator main window.



1.2 Jupyter Notebook Basics

It is recommended to become familiar with Jupyter basic concepts with the online documentation, <u>https://jupyter-</u>

notebook.readthedocs.io/en/stable/examples/Notebook/Notebook%20Basics.html

Some of the content is duplicated here for convenience. The online documentation should always be used as a reference.

1.2.1 The Notebook dashboard

When you first start the notebook server, your browser will open Notebook dashboard. The dashboard serves as a home page for the notebook. Its main purpose is to display the Notebooks and files in the current directory.

For example, here is a screenshot of the Jupyter dashboard. The top of the notebook list displays clickable breadcrumbs of the current directory. By clicking on these breadcrumbs or on sub-directories in the notebook list, you can navigate your file system.

Files Running Clusters	
Select items to perform actions on them.	Upload New - 2
	Name Last Modified File size
D 3D Objects	15 days ago
Contacts	15 days ago
Desktop	2 minutes ago
Documents	3 hours ago
Downloads	an hour ago
E Favorites	15 days ago
Links	15 days ago
C Music	15 days ago
New folder	3 hours ago
C node_modules	2 months ago
C OneDrive	8 months ago
C Pictures	6 days ago
Saved Games	15 days ago
Searches	15 days ago
Source	a month ago
Videos	15 days ago
D package-lock.json	2 months ago 616 B

1.3 Introduction to Jupyter Notebook GUI.

Jupyter Notebooks contain cells where you can either write code or markdown text. Notebooks contain multiple cells, some set as code and others markdown. Code cells contain code that can be executed live, and markdown contains text and images to explain the code.

Below image shows some options in a typical Jupyter Notebook. Individual cells can be executed by pressing on the RUN button as shown in the below image.

File Edit	View Insert Cell Kernel Widgets Help 2 € ↑ ↓ H Run ■ C → Markdown ▼ □	Trusted Python 3 O
Create new cell Blue band represents currently	Run selected cell Says if a particular cell is code or markdown This tutorial will guide you through the following steps: 1. Establish the communication between the host (PC) and the ECC608A 2. Generate MAC from device 3. Verify the MAC from device	
selected cell	 2.1. Establish the communication between the host (PC) and the ECC608A 1. Initialize the library which also loads it 2. Check if the correct device is connected to the PC. If the wrong device is connected, a ValueError exception will be raised. 	MARKDOWN CELL
In []:	2.1.1. Select the CryptoAuthLib Hardware Abstraction layer to KIT USB HID ATCA_KIT_I2C_IFACE = 1 cfg = cfg_atecx80a_kithid_default() cfg.detype = get_device_type_id('ATECC608A') cfg.cfg.atcahid.dev_interface = ATCA_KIT_I2C_IFACE cfg.cfg.atcahid.dev_identity = 0x6C #TFLXTLS device I2C address ATCA_SUCCESS = 0 print("Set devtype as ECC608A - Successfull")	CODE CELL
In []:	<pre>2.1.2. Establish the connection with the ECC608A-TLXTLS device on the USB dongle and che If the next Tutorial step yields an exception, the ECC608A is not correctly connected or of the wrong type. # Initialize interface assert atcab_init(cfg) == ATCA_SUCCESS, "Can't connect to the USB dongle" # Get connected device type info = bytearray(4) assert atcab_info(info) == ATCA_SUCCESS, "Can't read the ECC608A device information" dev_type = get_device_type_id(get_device_name(info)) # Checking if the connected device with selected device if dev_type = cfg_devtype: assert atcab_release() == ATCA_SUCCESS raise ValueError('Device not supported') putpt('Device initialization = Surgerfull")</pre>	eck its type

All cells in the Notebook can be executed in order by Kernel->Restart & Run All.

To run all cells in sequence.



2 Jupyter Notebook Tutorials

The TrustPlatform Design Suite comes with a Notebook Tutorials to easily prototype popular use cases for TrustFLEX and Trust&GO devices. Here are the available Jupyter Notebook Tutorials.

Jupyter Notebook Tutorials	Relative Path	Applicable devices
Manifest Generation	TNGTLS_Manifest_Generation\notebooks\TNGTLS Manifest File Generation.ipynb	Trust&GO
AWS IOT with TNG-TLS	TNGTLS_Use_Cases\notebooks\aws-iot\aws-iot with ECC608A-TNGTLS.ipynb	Trust&GO
Resource Generation	TFLXTLS_resource_generation\Crypto Resource Generator.ipynb	TrustFLEX
Accessory Authentication	TFLXTLS_Use_Cases\notebooks\accessory- authentication\ Accessory Authentication.ipynb	TrustFLEX
AWS Custom PKI	TFLXTLS_Use_Cases\notebooks\aws-iot\aws-iot with ECC608A-TLFXTLS.ipynb	TrustFLEX
Firmware Validation	TFLXTLS_Use_Cases\notebooks\secureboot\Firmware Validation with ECC608A-TFLXTLS Tutorial.ipynb	TrustFLEX
IP Protection	TFLXTLS_Use_Cases\notebooks\ipprotection\IP Protection with ECC608A-TFLXTLS Tutorial.ipynb	TrustFLEX
Secure Public Key Rotation	TFLXTLS_Use_Cases\notebooks\public-key- rotation\Public Key Rotation with ECC608A-TFLXTLS Tutorial.ipynb	TrustFLEX

3 Manifest Generation Notebook

Trust&GO device is one of the three devices available in the Trust Platform USB Dongle Board.

Trust&GO devices come with pre-programmed certificates in slots 10, 11 and 12, also slots 0-4 have pre-generated private keys, other than the previously mentioned slots all the other slots are locked.

The secure element manifest format is designed to convey the unique information about a device including its unique ID (e.g. serial number), public keys, and certificates. The manifest file generated can be used to register the device to cloud providers.

By default, Jupyter starts in Users directory (\$HOME for MacOS or Linux systems). For the remainder of this document, it will be assumed that the Trust_Platform folder is contained in the Documents folder.

Within the Jupyter dashboard, navigate to TNGTLS_Manifest_Generation\notebooks folder

Select the <u>TNGTLS Manifest File Generation.ipynb</u> notebook

💭 Jupyter	Quit	Logout
Files Running Clusters		
Select items to perform actions on them.	Upload	New - 2
□ 0	Last Modified	File size
	seconds ago	
E TNGTLS Manifest File Generation.ipynb	4 hours ago	7.4 kB
C common.py	a day ago	4.13 kB
manifest_generation_helper.py	a day ago	11.9 kB
a manifest_signing_helper.py	a day ago	5.03 kB

Run all cells of the TNGTLS_Manifest_Generation Notebook: Kernel->Restart & Run All

Note: Before executing the cells on Crypto Trust Platform, its required to have factory default program running on SAMD21 of Trust Platform. Refer to <u>4.1 CryptoAuth</u> <u>TrustPlatform Factory reset</u> section for reloading default program.



The Notebook will be used to generate a manifest file which can be uploaded into the public cloud provider of your choice (Google GCP, AWS IoT and soon to be supported Microsoft Azure). TNGTLS Manifest Generation notebook needs to be run for all Trust&Go example Notebooks that require a Manifest file.

If all the steps are run without errors, you will see two download links as shown below.

	2.2.3.1 Download Manifest
	Click the generated link to download the manifest
In [8]:	<pre>create_download_link(manifest_data, manifest_name, 'Download Manifest')</pre>
Out[8]:	Download Manifest
	2.2.3.2 Download Verification Certificate
	Click the generate link to download the manifest validation certificate
In [9]:	<pre>create_download_link(logger_cert, 'logger.pem', 'Download Logger Certificate')</pre>
Out[9]:	Download Logger Certificate
logger.pem	↑ 🚺 01236cd3feb4994json ↑

Click on "Download Manifest" and "Download Logger Certificate" to download the manifest and logger file.

4 Loading Manifest to AWS-IoT

This hands-on lab is intended to demonstrate how to load a manifest file into AWS-IOT to enable device connectivity.

We would be using the manifest file and logger file generated in the TNGTLS Manifest File generation notebook. The Manifest file contains information about the device including serial number, public keys and certificates.

Loading a manifest file to AWS_IOT through Jupyter Notebook:

1. From the Jupyter Home page, navigate to **TNGTLS_Use_Cases/notebooks/aws**iot/aws-iot with ECC608A-TNGTLS.ipynb notebook file and open it.

🔁 jupyter	Quit Logou
Files Running Clusters	
Select items to perform actions on them.	Upload New -
0 V Documents / usecasetool / TNGTLS_Use_Cases / notebooks / aws-iot	Name Last Modified File size
	seconds ago
aws-iot with ECC608A-TNGTLS.ipynb	5 minutes ago 5.24 k
C1 MicrochipManifestHandler.py	a day ago 8.57 k
README.md	a day ago 838

Opening the notebook from Jupyter home page should load the following on the browser.

ments/usecaseto	I/TNGTLS_Use_Cases/notebooks/aws-iot/aws-iot%20with%20ECC608A-TNGTLS.ipynb	
File	Edit View Insert Cell Kernel Widgets Help	Trusted Python 3 O
+	8< 41 K + + H Run ■ C >> Markdown • □	
	d haden des Barr	
	1.Introduction	
	This Notebook tutorial will allow you to load a TNG manifest into aws-iot to enable device connectivity	
	1.1 Tutorial Prerequicites	
	1.1. Iutorial Prefequisites.	
	The following code is runs on python 3.x environment. This step of the tutorial will attempt to load all the necessary me machine. If the above modules are already installed you can safely step over the next Tutorial step.	odules and their dependencies on your
	pip install -U module_name	
	<pre>In []: !python -m pip install -r///requirements.txt</pre>	
	112 Configure AWS CLI and environment	

2. This notebook requires user input in some of the intermediary steps so Run All option in Jupyter is not recommended. Run steps 1.1 and 1.2, these steps will install all the modules required for the Notebook and import the required modules.

- 3. Run step 1.3, under step 1.3 it will prompt you to enter the following details
 - a. Access key
 - b. Secret key
 - c. Region name

These details will be used to setup AWS-CLI in your PC. You can get these details from your AWS account.

The output after entering all the details will look like the image below. Credentials used in the below image should not be used, you are needed to enter the credentials tied to your own account.

In [8]:	access_key = i configure_aws_ secret_key = i configure_aws_ region = input configure_aws_	<pre>input('Enter Access key\n\r access_key(access_key) input('Enter Secret access secret_access_key(secret_k c('Enter region\n\r') _configure_region(region)</pre>	') <mark>key\n\r'</mark>) ey)				
	Enter Access k	cey (
	AKIAQAL5EMPTMC Setting aws ac Done	:VSHGMN :cess key					
	Enter Secret access key						
	gM4oKuVI9vLvqw48IJKG7tUu/GmQ1u2jTcbjQtqy Setting aws secret access key Done						
	Enter region						
	cn-north-1 Setting aws re Done	gion					
	Name	Value	Type	Location			
	profile access_key secret_key region	<pre><norm <="" <norm="" norm="" norm<="" td=""><td>None ed-credentials ed-credentials config-file</td><td>None 5-file 5-file ~/.aws/config</td><td></td><td></td><td></td></norm></pre>	None ed-credentials ed-credentials config-file	None 5-file 5-file ~/.aws/config			

4. Run step 1.4, it will create Upload button. Press on that button, it will open file explorer window, there you need to navigate and choose the manifest file generated using TNG Manifest Generation Notebook.

<pre>In [10]: manifest = FileUpload(accept-'*.json', multiple display(manifest)</pre>	e-False)
Open	×
→ ↑ ↑ → This PC > Downloads	✓ ປັ Search Downloads
rganize 🔻 New folder	■ • Ⅲ ?
 Quick access Desktop * Downhoads * Downhoads * Downhoads * Dit236cd3feb4994 D1236cd3feb4994 D1236cd3feb4994	
File name: 01236cd3feb4994b01_manifest.json	All Files (*.*)

5. Run step 1.5, it will create Upload button. Press on that button, it will open file explorer window, there you need to navigate and choose the validation certificate file generated using TNG Manifest Generation Notebook.

<pre>In [6]: validator = FileUpload(accept='*.pe display(validator)</pre>	', multiple=False)		
Open			
→ * ↑ ↓ > This PC > Downloads	~ Ö	Search Downloads	م
rganize 👻 New folder			- 💷 🚱
 Downloads Documents Pictures My Files (E) Bin Personal validate Z SPG Apps This PC Do Objects Desktop 			
File name: logger.pem	~	All Files (*.*) Open	∨ Cancel

6. Run step 1.6, the successful completion of the step will import the certificate.

<pre>manifest_data = json.loads(manifest.data[0]) validation_certificate = validator.data[0] invoke import manifest('Default' _ manifest data_ validation_certificate)</pre>
invoke_import_manifest(berault, manifest_uata, valuation_tertificate)
number of certificates: 1
Loading the manifest item
uniqueId: 01232d76543e3c1401
About to try certificate import
Response: {'ResponseMetadata': {'RequestId': 'c60038f5-5fbf-406b-b4d8-06ab5bdca2dc', 'HTTPStatusCode': 200, 'HTTPHeaders': {'da
te': 'Thu, 26 Sep 2019 19:46:49 GMT', 'content-type': 'application/json', 'content-length': '209', 'connection': 'keep-alive',
x-amzn-requestid: <pre>c60038f5-5fbf-406b-04d8-06ab5bdca2dc</pre> , access-control-allow-origin: **, x-amz-apigw-id: Apb5cE8MPHcF Apple - Apple
WUG= , X-amin-trace-ii : KOOt=1-50801549-Cet090aCde5880T4/DD0C354 }, KetryAttempts : 0}, CertItLateArn : arn:aws:10t:US-W pct_3-05206500AAA4:cont/600231hEdf66553200-500Af170Ec3F44aacF546f4326F320Adba4743 'contificatad': 'add0002-750934hEdf
66557388cf986617953bfd4a57b306f7330dbd474a')
Certificate import complete - returning
MANIFEST_IMPORT_SUCCESS_arn:aws:iot:us-west-2:257966804464:cert/e0d8082c7b88341b5df665f2308cf98461795c3bfd4ae57b306f7239dbda474
a arn:aws:iot:us-west-2:257966804464:thing/01232d76543e3c1401

7. Run step 1.7, successful execution of this step verifies that the manifest was successfully uploaded, and it outputs the corresponding unique ID

In [5]: manifest_data = json.loads(manifest.data[0]) validation_certificate = validator.data[0] invoke_validate_manifest_import(manifest_data, validation_certificate)

number of thingIds to check: 1

Checking the manifest_item uniqueId: 01232d76543e3c1401 Manifest was loaded successfully

4.1 CryptoAuth TrustPlatform Factory reset

Once any of the embedded project is loaded to CrytoAuth TrustPlatform, the default program that enables interaction with TrustPlatform tools will be erased.

Before using the Platform with any other notebook or tools on PC, its required to reprogram the default .hex file. Default hex file is available at

TFLXTLS_Use_Cases\c\Factory_Program.X\CryptoAuth_Trust_Platform.hex

To reprogram using Atmel Studio:

- 1. Navigate to AtmelStudio -> Tools -> Device Programming
- 2. Select Tool as nEDBG and Apply
- 3. Go to Memories and navigate to above path under Flash dropdown
- 4. Check both Erase Flash and Verify Flash
- 5. Click on Program

Tool Device	Interface Device signature Target Voltage 18A SWD ~ Apply 0x1001030A Read 3.3 V Read
Interface settings Tool information Tool settings	Device Erase Chip × Erase now Flash (256 KB) E:\New folder\usecasetool\TFLXTLS_Use_Cases\c\Factory_Program.X\CryptoAuth_Trust_Platform.hex ×
Memories Fuses	 ✓ Erase Flash before programming ✓ Verify Flash after programming ✓ Advanced
Security	User Page (256 bytes) ✓ ✓ Erase User Page before programming ✓ Verify User Page after programming ✓ Advanced

To reprogram using MPLAB:

- 1. Open TFLXTLS_Use_Cases\c\Factory_Program.X project in MPLAB IDE
- Program the Crypto Trust platform by navigating to CryptoAuth_Trust_Platform_Factory_Program -> Make and Program Device

Now, Crypto Trust Platform contains factory programmed application that enables interactions with Notebooks and/or PC tools.

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as

a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

 Product Support – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived

software

- **General Technical Support** Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- Business of Microchip Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support.

Local sales offices are also available to help customers. A listing of sales offices and locations is included

in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of
 these methods, to our knowledge, require using the Microchip products in a manner outside the
 operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is
 engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.

• Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be

а

violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL,

STATUTORY

OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq,

Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB,

OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, SAM-BA, SpyNIC, SST,

SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology

Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight

Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming.

ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient

Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE,

Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademark of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2018, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN:

Quality Management System Certified by DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California

and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.



Worldwide Sales and Service

AMERICAS

Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 **Technical Support:** http://www.microchip.com/ support Web Address: www.microchip.com Atlanta Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 Austin, TX Tel: 512-257-3370 Boston Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 Chicago Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 Dallas Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 Detroit Novi, MI Tel: 248-848-4000 Houston, TX Tel: 281-894-5983 Indianapolis Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 Los Angeles Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 Raleigh, NC Tel: 919-844-7510 New York, NY Tel: 631-435-6000 San Jose, CA Tel: 408-735-9110 Tel: 408-436-4270 Canada - Toronto Tel: 905-695-1980 Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney Tel: 61-2-9868-6733 China - Beijing Tel: 86-10-8569-7000 China - Chengdu Tel: 86-28-8665-5511 China - Chongqing Tel: 86-23-8980-9588 China - Dongguan Tel: 86-769-8702-9880 China - Guangzhou Tel: 86-20-8755-8029 China - Hangzhou Tel: 86-571-8792-8115 China - Hong Kong SAR Tel: 852-2943-5100 China - Nanjing Tel: 86-25-8473-2460 China - Qingdao Tel: 86-532-8502-7355 China - Shanghai Tel: 86-21-3326-8000 China - Shenyang Tel: 86-24-2334-2829 China - Shenzhen Tel: 86-755-8864-2200 China - Suzhou Tel: 86-186-6233-1526 China - Wuhan Tel: 86-27-5980-5300 China - Xian Tel: 86-29-8833-7252 China - Xiamen Tel: 86-592-2388138 China - Zhuhai Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore Tel: 91-80-3090-4444 India - New Delhi Tel: 91-11-4160-8631 India - Pune Tel: 91-20-4121-0141 Japan - Osaka Tel: 81-6-6152-7160 Japan - Tokvo Tel: 81-3-6880- 3770 Korea - Daegu Tel: 82-53-744-4301 Korea - Seoul Tel: 82-2-554-7200 Malaysia - Kuala Lumpur Tel: 60-3-7651-7906 Malaysia - Penang Tel: 60-4-227-8870 Philippines - Manila Tel: 63-2-634-9065 Singapore Tel: 65-6334-8870 Taiwan - Hsin Chu Tel: 886-3-577-8366 Taiwan - Kaohsiung Tel: 886-7-213-7830 Taiwan - Taipei Tel: 886-2-2508-8600 Thailand - Bangkok Tel: 66-2-694-1351 Vietnam - Ho Chi Minh Tel: 84-28-5448-2100

EUROPE

Austria - Wels Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 Denmark - Copenhagen Tel: 45-4450-2828 Fax: 45-4485-2829 Finland - Espoo Tel: 358-9-4520-820 France - Paris Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 France - Saint Cloud Tel: 33-1-30-60-70-00 Germany - Garching Tel: 49-8931-9700 Germany - Haan Tel: 49-2129-3766400 Germany - Heilbronn Tel: 49-7131-67-3636 Germany - Karlsruhe Tel: 49-721-625370 Germany - Munich Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 Germany - Rosenheim Tel: 49-8031-354-560 Israel - Ra'anana Tel: 972-9-744-7705 Italy - Milan Tel: 39-0331-742611 Fax: 39-0331-466781 Italy - Padova Tel: 39-049-7625286 Netherlands - Drunen Tel: 31-416-690399 Fax: 31-416-690340 Norway - Trondheim Tel: 47-7289-7561 Poland - Warsaw Tel: 48-22-3325737 Romania - Bucharest Tel: 40-21-407-87-50 Spain - Madrid Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 Sweden - Gothenberg Tel: 46-31-704-60-40 Sweden - Stockholm Tel: 46-8-5090-4654 UK - Wokingham Tel: 44-118-921-5800 Fax: 44-118-921-5820