TECHNICAL BRIEF:
PGP® COMMAND LINE FROM SYMANTEC™

PGP Command Line from Symantec

Who should read this paper

Technical Information Technologists

Symantec

Content

Executive Summary

Data transfer and processing systems form the circulatory system of most organizations, exchanging large volumes of information between internal systems, suppliers, and customers. But legacy data transfer and processing systems are especially prone to security breaches because traditional file transfer and email protocols have no built-in security.

For organizations that must securely exchange large volumes of information, PGP® Command Line from Symantec™ can protect business-critical data easily and with little impact on existing systems. Command Line can also be used to protect large volumes of information stored on servers and backup media from unauthorized access.

This Technology Overview presents examples of ways that Command Line can be used to encrypt data in automated business processes. The white paper is intended for IT managers and technical implementers who are responsible for developing, managing, and securing business processes. Sample scripts show how easily organizations can integrate Command Line. If you have more complex requirements, Symantec and its partners can help you plan the technology upgrade and guide you through the process.

Choosing the Right Solution

Automated business processes that store and forward critical information are becoming more and more risky. Malicious code, hacks, and internal compromises can quickly turn a corporate asset into a liability.

Whether mishandled, lost, stolen, or intercepted, data can become your worst enemy. Lost backup tapes, stolen computers, and misused privileges all represent common data security breaches. The rise in identity theft has turned these breaches from purely internal matters into incidents with significant financial and legal ramifications. Breaches are often widely publicized, hurting the organization's reputation. And the average cost of a compromised record is $145, according to a Ponemon Institute study http://www.ponemon.org/news-2/58

In other words, a privacy breach will reduce your organization's profits, may cost you your job, and could even mean the end of your organization itself.

The risk of security breaches has led many organizations to reconsider how they handle data in transit and at rest.

Data: Always In Transit

Data is often described as in transit or at rest; however, this categorization is less than perfect. Data is almost always in transit, whether transferred via FTP over the Internet, stored in a storage area network (SAN), or archived on a backup tape in a delivery truck en route to offsite storage. Data commonly referred to as "at rest" is often actually waiting to be transferred.

Different means of encryption have commonly been used to protect data in transit and data at rest. These methods include session and file encryption.

Session Encryption

To protect data being transferred over networks, temporary encryption keys used only for the session are generated and used to encrypt a transfer from the origin to the destination. Common session encryption technologies include IPSec or SSL VPN connections, SSH or SFTP network transfers, and HTTPS Web-based transfers. Although the data is protected in transit with these methods, it remains unencrypted

before and after transfer, presenting a potential target for a breach. Another risk of session encryption is that temporary files and backups may still be found on a disk drive, even if deleted.

File Encryption

The alternative to session encryption is to encrypt the data at rest instead of in its transit session. In other words: encrypt the files, not the transmission, to better protect it from compromise in the event of accidental loss or theft. Common strategies include file encryption (or archive encryption) that uses OpenPGP, PKCS#7, or proprietary password encryption. File encryption secures data both at rest and in transit, which safeguards the information against a breach of the servers and interception over the wire.

A Standard Solution Approach

We have shown that file encryption is superior to session encryption because it protects the data in more circumstances. This is why the security market offers a multitude of file encryption solutions. To choose the file encryption solution that is best for your organization, evaluate the various offerings on how well they fulfill these four major requirements:

- **Support standards-based encryption and file formats** – Proprietary formats hinder the broad acceptance of encryption. Having a standards-based solution ensures that you can securely exchange information with present and future partners, and that you can still access archived data for many years to come.
- **Easily integrate with existing processes** – Your organization may use a variety of applications to manage and process sensitive information. Choose an encryption solution that is flexible enough to integrate with both new and legacy applications.
- **Support a broad set of platforms** – Your organization may use a variety of digital platforms that process sensitive information. Choose an encryption tool that supports a heterogeneous set of platforms and operating systems, especially if the applications that you use run on systems as diverse as Windows servers, UNIX workstations, and midrange or mainframe systems. You should also consider which platforms you may need to support in the next five years.

Provide advanced key management – To protect private keys and preserve access to encrypted data, the encryption solution must include advanced key management technologies such as central key storage and key splitting. Central key storage lets you avoid having to touch each system when keys change or additional servers join the system; storing keys centrally is especially important if your encryption solution connects several systems. Key splitting controls the access to and use of private keys for operational security. It is often used to protect critical non-personal keys for corporate access, such as archiving, e-discovery, or data recovery. With key splitting, a number of authorized key holders each receive a key share. A minimum number of key shares, also called a “threshold,” must be met to reconstitute a key and make it available for use. Other advanced key management technologies that you may require in your solution include methods for ensuring corporate access to encrypted data if required by policy or regulatory mandates, even in the event that a private key is lost.

Command Line

Command Line is a file encryption solution that fulfills all four of the requirements and is designed for flexibility. It is ideal for use with batch processing, network transfer, and backup applications.

Standards-based encryption and file formats. Command Line uses standards-based OpenPGP (IETF RFC 2440) cryptography to compress, encrypt, and digitally sign files and directories. The software also encrypts emails in OpenPGP and S/MIME format. Built on the PGP® Software Development Kit (PGP® SDK), Command Line uses the same core cryptographic libraries that are built into other Symantec Encryption products. Command Line also supports commonly used file compression methods: Zip, BZip2, and ZLib.

Data encrypted with Command Line can be decrypted by using other Command Line clients or Symantec Encryption Desktop software. For users without Command Line or Symantec Encryption Desktop software, Command Line can generate Self-Decrypting Archives (SDAs). SDAs are archives encrypted with a passphrase that can be opened by users without Symantec software. Because SDAs use symmetric encryption, the encryption passphrase must be communicated to the intended recipient "out of band", for example by phone, fax, or short message service (SMS). With Command Line, SDAs can be created for execution on any supported platform, allowing encrypted files to be easily transferred for use on both desktop and server platforms (for example, by creating an SDA on Linux to be decrypted on Windows).

Easy applications integration. Command Line runs as a shell-based executable. Command Line is accessible from a variety of scripting languages, including UNIX scripts, Windows batch scripts, PERL, and other scripting tools and applications that can call an executable and pass arguments. This functionality allows Command Line to be easily integrated into a wide variety of applications, such as enterprise backup applications.

Broad platform support. Command Line is available on a broad range of enterprise server platforms. In addition to these platforms, any version of Command Line can be used to create SDAs that run on another supported platform (for example, an SDA created on AIX runs and decrypts on Windows Server). Command Line is supported on Windows, Mac, Linux, HP-UX, AIX, and Solaris operating systems.

Advanced key management. Command Line enhances private key security by supporting key splitting. Additionally, Command Line ensures long-term accessibility to encrypted data with Additional Decryption Key (ADK) technology. Command Line can associate ADKs with PGP keys at the time of original key generation. When information is encrypted to a PGP key with an assigned ADK, Command Line will also encrypt information to the ADK. In the event that a private key is lost or access to encrypted data is required by policy or regulations, an ADK can regain access to and decrypt information.

Real-life Use Cases

The following concise examples show how some customers in the financial, health care, and services industries use Command Line. Many of these customers use Command Line as well as other Symantec Encryption solutions.

ACS

Customer confidence and regulatory compliance are essential to the success of Affiliated Computer Services, Inc. (ACS). A Fortune 500 business process and information technology outsourcer, ACS handles high volumes of sensitive corporate and customer data for clients in more than 100 countries. To provide additional security, ACS purchased licenses of Command Line for 150 servers to secure communication between systems.

Bertelsmann

A global media company with 97,000 employees in 60 countries, Bertelsmann needed a scalable, cost-effective encryption solution to protect sensitive data and comply with national and regional data privacy laws. As the foundation of its enterprise data protection strategy, Bertelsmann chose Symantec Encryption products to deliver encryption across the enterprise. The Bertelsmann subsidiary Bookspan, a U.S. book club, uses Command Line to protect its file transfers with partners.

DeKalb Medical Center

DeKalb Medical Center must comply with federal regulations designed to protect the privacy of patient records. As part of its enterprise data protection strategy, DeKalb Medical Center chose the Symantec Encryption products to meet all of its encryption needs. DeKalb Medical Center decided to phase out its VPN solution for its FTP server transmissions with partners and replace it with Command Line encryption.

PHNS

A business process outsourcer for health care providers, PHNS needed an enterprise data protection strategy to help comply with industry and government regulations protecting patient privacy and financial records. Command Line protects confidential server-to-server communications in back-end patient record and financial management applications.

Rule Financial

With customers throughout the United Kingdom and Europe, Rule Financial needs to protect sensitive data and comply with relevant industry regulations. The financial services company selected Command Line to secure transactions between banks and brokers. Symantec Encryption now forms the core of Rule Financial's enterprise data protection strategy to defend customer and business partner data — wherever it goes.

Securing Automated Business Processes

Because Command Line is a scripting and shell-based encryption application, it can integrate quickly with both off-the-shelf applications and custom scripts. Command Line also provides the advanced key management options that enterprises require for critical automated business process applications, such as securing multisite FTP transfers and encrypting backup tapes for offsite storage.

To illustrate how Command Line meets multiple transfer, storage, and backup encryption requirements, the following scenario presents an example of a mid-sized business with a variety of encryption requirements. This hypothetical example of "GlobalCPG Corporation" includes the experiences of real-life Symantec customers, without revealing any customer's confidential encryption strategies, policies, or procedures.

GlobalCPG Corporation

GlobalCPG Corporation is a midsized electronic consumer goods manufacturing company with 750 employees. As a subsidiary of a publicly traded conglomerate, GlobalCPG must meet the same stringent reporting and compliance requirements as its parent company. GlobalCPG has customers and distributors throughout the world, and it must protect both business and individual data. GlobalCPG has recently begun to develop a customer relationship management (CRM) system that tracks consumers to help it better understand consumer satisfaction and preferences.

GlobalCPG decided to adopt data encryption technology to address regulatory compliance and protect its sensitive corporate and customer data, even in the event of loss or theft. With three business applications and processes to secure, GlobalCPG deployed Command Line in these ways:

- **Daily transmission to external trading partners** – Encrypt EDI data transmissions for supply-chain integration.
- **Internal network transfer of daily financial results** – Encrypt data exchange between internal heterogeneous systems.
- **Tape backup encryption** – Encrypt individual files by using a split PGP key.
- **Data distribution to partners without encryption software** – Create a Self-Decrypting Archive on IBM AIX to run and decrypt on Windows machines without Command Line installed.

Daily File Transmission to External Partners

GlobalCPG tightly integrates its manufacturing supply chain through daily Electronic Data Interchange (EDI) with its trading partners to order shipments of raw material and parts. The EDI data is generated on a Windows server, where it is encrypted and copied to a file transfer server that sends the files to the trading partners via FTP. The entire process is fully automated. GlobalCPG chose to encrypt the data in the OpenPGP format because OpenPGP is a widely accepted, easy-to-implement industry standard.

Command Line Integration

Figure 1 illustrates the role of Command Line in the EDI supply chain application processing. Following successful transfer, the encrypted files will be securely deleted.

Figure 1: Encrypting EDI data for transmission to trading partners

Scripting

The following script calls illustrate the use of Command Line to encrypt files with the OpenPGP standard and perform secure deletion.

Pre-backup encryption

```
PGP  --e ~/edi_data/*xml -r "Trading Partner ABC Corporation – ERP" –o ABC_EDI.pgp
```

Encrypt all XML files in temporary Finance data directory	Encrypt to trading partner ABC Corporation's ERP system key	Specify output archive filename

After the encrypted files are transferred, a subsequent Windows batch script calls Command Line to perform a secure wipe of all temporary files used for the transfer: the XML data files and the encrypted file.

Post-backup file wipe

```
Pgp –wipe *xml *pgp --wipe-passes 5
```

Initiates secure file deletion	Wipes all temporary and output files	Performs 5 wipe passes, exceeding military-grade requirements for secure file deletion

After receiving the encrypted files, ABC Corp. will route the encrypted XML files to an ERP system. The system will use Command Line to decrypt the files temporarily for processing, and to subsequently perform a secure wipe of the decrypted files.

Internal Network Transfer of Daily Financial Results

At the end of each business day, all subsidiaries of GlobalCPG's parent company transfer details of the day's business. This data is used to create an executive dashboard and monitor large customer accounts laterally across subsidiaries. The data source and target systems run on different platforms, including Windows and UNIX systems. Although the FTP transfers are made over a VPN connection, the data sets are used by the sales and finance departments and remain on the departmental servers until removed at the end of each quarter. Because financial information is transferred between departments and stored on systems for months, encrypting the data ensures that only authorized applications or administrators have access to it before GlobalCPG's parent company reports financial results. Encrypting this data is part of the compliance programs at GlobalCPG and its parent company.

Command Line Integration

When integrating Command Line, GlobalCPG considered and implemented these two requirements:

- Multiple files should be compressed and stored in a single encrypted archive.
- Following successful transfer, the encrypted files should be securely deleted.

Figure 2: Encrypting daily financial results for corporate parent

To create a single archive, Command Line's PGP Zip function stores files and directories in a single encrypted archive with commonly used compression. Command Line supports encryption and decryption of PGP Zip archives, as does Symantec Encryption Desktop.

Scripting

The following script calls illustrate the use of Command Line to encrypt files in a PGP Zip archive and perform secure deletion.

Pre-transfer encryption

```
PGP  --e ~/finance_data/*xml -r "Parent – Sales" –r "Parent – Finance" –o
upload.pgp --archive
```

Encrypt all XML files in temporary Finance data directory	Encrypt to both the parent corporation's Sales and Finance keys	Specify output archive filename	Create the archive as a PGP Zip file

After the encrypted files are transferred, a subsequent UNIX Shell Script calls Command Line to perform a secure wipe of all temporary files used for the transfer: the XML data files and the encrypted PGP Zip file.

Post-transfer file wipe

```
pgp –wipe *xml *pgp –wipe-passes 5
```

Initiates secure file deletion	Wipes all temporary and output files	Performs 5 wipe passes, exceeding military-grade requirements for secure file deletion

After receiving the encrypted files, ABC Corp. will route the encrypted XML files to an ERP system. The system will use Command Line to decrypt the files temporarily for processing, and to subsequently perform a secure wipe of the decrypted files.

Tape Backup Transport to Offsite Storage

Each week, GlobalCPG sends a backup of databases running on the AIX platform to an offsite storage facility. This process is part of the organization's business continuity and compliance programs. In the hours before the weekly tape backup, database data is prepared for backup, generating large database files stored in a staging directory. The contents of this staging directory are then transferred to tape. The entire process is automated using a UNIX shell script.

Command Line Integration

When integrating Command Line, GlobalCPG considered and implemented these three requirements:

- Database backup files must be encrypted individually.
- Following successful tape backup, all temporary files must be securely deleted.

Decryption of encrypted backups requires key splitting among at least two of the five IT administrators who are authorized to request retrieval of backups from the offsite storage vendor.

Figure 3: Encrypting tape backups for offsite storage

When performing encryption, Command Line will by default encrypt individual files and output a new encrypted file with the .pgp extension. Encrypting to a split PGP key does not require special configuration; however, during decryption, the prerequisite number of key shares must be available to reconstitute the key and perform decryption.

Scripting

The following script calls illustrate the use of Command Line to create split keys, encrypt files, perform secure deletion, and decrypt files using a split PGP key.

Split tape backup encryption key

Initiate a key split operation for the tape backup key

Set threshold of 2 keys

Create a share each for Admins 1 & 2

```
pgp --split-key "GlobalCPG Corp DB Tape Backup" --threshold 5 --share "1:Admin1" -
-share "1:Admin2" --share "1:Admin3" --share "1:Admin4" --share "1:Admin5" --
passphrase k49cxk5 –force
```

Create a share each for Admins 3, 4, & 5

Provide backup key passphrase and authorize split

Five administrators are provided with one key share each. With a threshold for reconstitution of two key shares, two administrators will be required to authorize decryption using GlobalCPG's tape backup encryption key.

Pre-backup encryption

```
pgp --e ~/db_backup/* --recipient "GlobalCPG Corp DB Tape Backup"
```

Encrypt all files in temporary backup directory

Encrypt to GlobalCPG's tape backup encryption key

Post-backup file wipe

```
pgp --wipe *csv *exe --wipe-passes 5
```

Initiates secure file deletion

Wipes all temporary and output files

Performs 5 wipe passes, exceeding military-grade requirements for secure file deletion[7]

Once backups are committed to tape, they are stored and then transferred by a delivery agent to an offsite storage facility. When a backup tape is needed, it is delivered to GlobalCPG. The needed backup files are copied from the tape and then prepared for decryption by authorized administrators.

Decryption with split keys

The third administrator authenticates

```
pgp --cache-passphrase "Admin3" --passphrase b6s3v2 --passphrase-cache
```

Cache the passphrase of the third administrator

Provide the passphrase

Enable passphrase caching

The fifth administrator authenticates

```
pgp --cache-passphrase "Admin5" --passphrase 8gmas2 --passphrase-cache
```

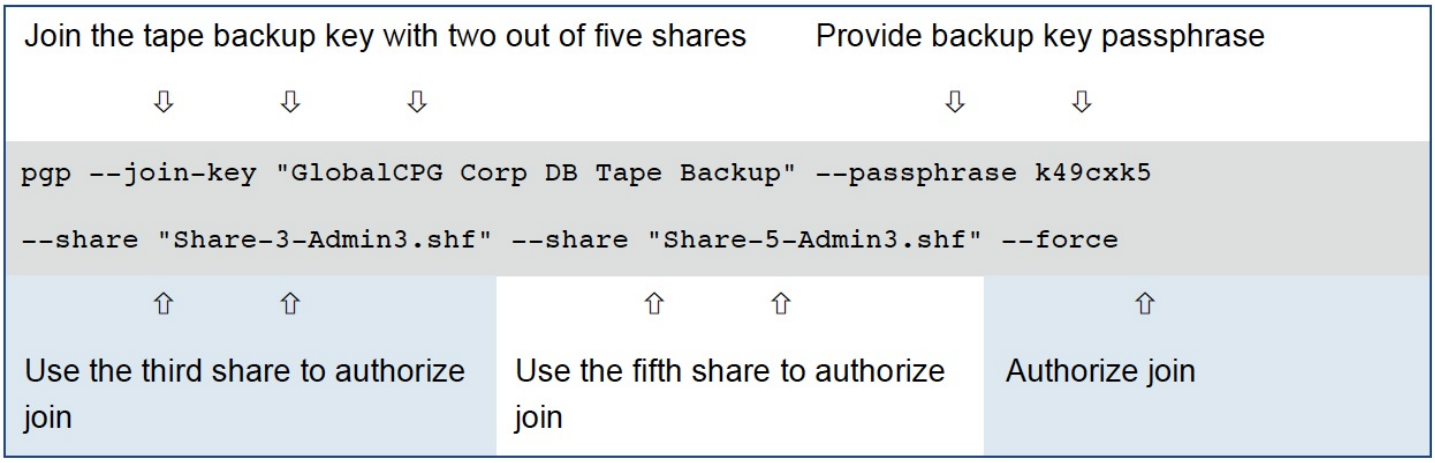
Cache the passphrase of the fifth administrator

Provide the passphrase

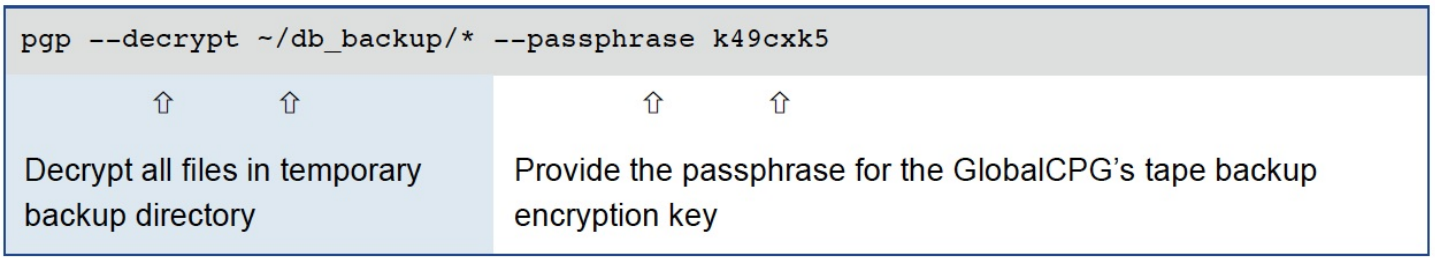
Enable passphrase caching

After each administrator provides the passphrase to his/her private key, key reconstitution can be performed and tape backups recovered.

Recover Tape Backup Key



Decrypt Backups



Once the tape backup encryption key is reconstituted, it can be used immediately for decrypting backups.

Data Distribution to Partners without Encryption Software

GlobalCPG has outsourced the sales and claims functions for its extended warranty program to a third party. This warranty service bureau contacts new customers, sells extended warranties, and settles claims. GlobalCPG exports records of new customers from its mainframe system and delivers them to the warranty service bureau on a DVD in various CSV files, which are simple text-based files that are easily imported into a wide range of applications.

GlobalCPG uses Command Line encryption to protect customer data both in transit and when not in use. Because the service bureau does not have a Command Line license, GlobalCPG creates an SDA that can be decrypted without the use of Symantec Encryption software.

Command Line Integration

When integrating Command Line, GlobalCPG considered and implemented these three requirements:

- Multiple files should be stored in a single encrypted archive.
- The SDA generated on IBM AIX must be executable on the service bureau's Windows 8 systems.

Figure 4: Encrypting customer lists for use by business partner

Command Line SDAs can be generated for any of the platforms supported. Instead of using asymmetric encryption, Command Line SDAs use passphrase-based symmetric encryption that requires the passphrase to be shared with the authorized recipient(s) to allow decryption. GlobalCPG shares the decryption passphrase "out-of-band" during a phone conversation with the service bureau rather than delivering the passphrase via the same means as the physical media, eliminating a potential risk.

Scripting

The following script calls illustrate the use of Command Line to encrypt files into a Windows SDA.

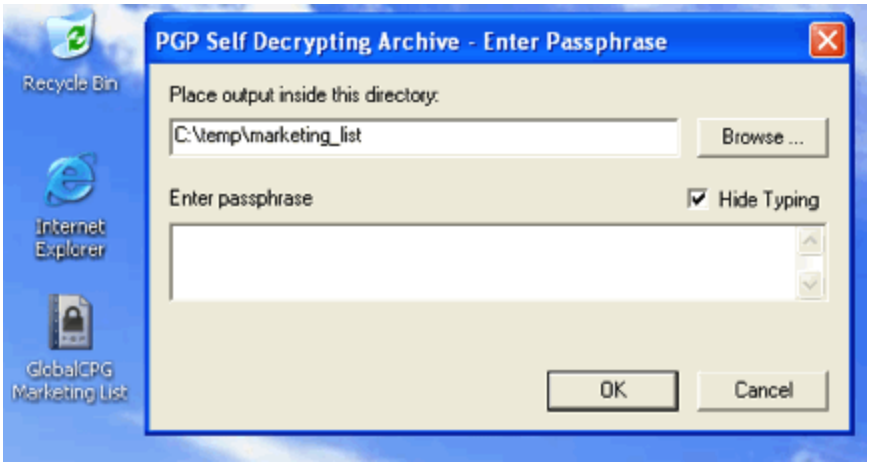
Creating SDAs

```
Pgp --e *.csv –sda -o csvs.exe --symmetric-passphrase "sdf@3r4*@dJ" --target-
platform win32
```

Perform encryption of all CSV files	Create a Self-Decrypting Archive (SDA) named 'csvs.exe'	Set the passphrase for encryption to "sdf@3r4*@!dJ"	Create an executable archive for Microsoft Windows

The customer lists are encrypted and packaged into a PGP SDA and output as a Windows EXE.

Example of Decryption on Windows



The recipient of the encrypted SDA launches the file on Microsoft Windows. The shared passphrase (the same passphrase used to encrypt the archive) is then used to decrypt the SDA and the encrypted CSV files.

Conclusion

Adding encryption to business applications and processes allows organizations to address risk mitigation, compliance, and the potential consequences of a security breach. With Command Line, integrating encryption is a matter of adding a few lines of command line calls. Most importantly, Command Line addresses four critical requirements for adding encryption to critical processes:

- Standards-based encryption
- Easily integrated
- Broad platform support
- Advanced key management

From tape backup to batch FTP transfers and distribution of sensitive materials to partners, Command Line provides enterprises with robust encryption capabilities for their automated data processing applications. Learn more about Command Line and other PGP encryption solutions by contacting a Symantec representative. Additional information is also available on the Symantec website: http://www.symantec.com/encryption/

About Symantec

Symantec is a global leader in providing security, storage and systems management solutions to help consumers and organizations secure and manage their information-driven world. Our software and services protect against more risks at more points, more completely and efficiently, enabling confidence wherever information is used or stored.

For specific country offices and contact numbers, please visit our website.

Symantec World Headquarters
350 Ellis St.
Mountain View, CA 94043 USA
+1 (650) 527 8000
1 (800) 721 3934
www.symantec.com

2/2015 21347937