STANDARD OPERATING PROCEDURE
Do not Photocopy

Document Information Classification: Unrestricted

Title:	**Secure Deletion of TRE Data**
Effective Date:	**24 Feb 2020**
Reference Number:	**SOP-05-15**
Version Number:	**1.7**
Owner:	**Information Security Manager,**
Review Date:	**24 Feb 2022**

Table of Contents

1. Purpose

This document provides guidance for the secure deletion of datasets from the TRE.

In the context of the TRE, *deletion of a dataset* refers to the technique of ensuring there is no possibility of further access to that asset, and being able to provide evidence of this deletion to the Data Controller who originally provided the dataset.

The TRE, and all systems contained within, is involved in handling information that must be managed in a way that ensures its confidentiality, availability and integrity. Implementing security controls throughout the lifecycle of a system can help the TRE achieve its ISMS objectives, regulatory requirements and the needs of its users.

2. Scope

All research datasets that are transferred into a project's file system within the TRE.

3. Responsibilities

The TRE Operations Manager is responsible for:

- Ensuring the data deletion requirements are recorded and understood at the project application stage
- Providing this SOP to a Data Controller if they require details of the TRE data deletion methods
- Providing evidence to a Data Controller that the exact same datasets originally transferred into the TRE have been deleted
-

The TRE System Administrator is responsible for:

- Performing the data destruction techniques as prescribed by this SOP
- Generating and storing data destruction reports in a format that can be shared with Data Controllers

4. Procedure

4.1. Background

It is important to note the significant difference between data deletion and data destruction*.

For most operating systems, file deletion only moves the file into a 'recycling bin'. Even emptying the recycling bin usually only removes the file's entry from the file-system's index (e.g. the File Allocation Table in Windows). This means it might still be possible to recover the original file using data recovery tools.

Data destruction is a different technique, and as it is more rigorous, it takes longer.

This document only describes the process of data destruction.

**Within the context of this SOP, data destruction and secure deletion are synonymous.*

4.2. Levels of Data Destruction

This document describes two methods for destroying data: destruction of files and destruction of encrypted volumes.

4.2.1. File Destruction

This method involves the destruction of files individually from the filesystem. When data is removed using file destruction techniques, the TRE Service cannot make a guarantee that the data wouldn't be recoverable if specialist disk scanning tools were used. But from a practical perspective, it would be exceptionally difficult for a user of the TRE service to recover any files removed using this method, and therefore it is suitable if the Data Controller requests removal of access to certain parts of the project's data. An applicable scenario would be if the data controller suddenly finds out some the data imported into a live project's working area within the TRE was not anonymized to the required level, and therefore the relevant files must be destroyed. In this scenarios, destruction of the entire data volume is not possible as the project still requires access to other data.

4.2.2. Data Volume Destruction

Each TRE Project stores its data in an encrypted volume that cannot be accessed outside the boundaries of the TRE Project. Each volume requires a unique key file for decryption, and this key file can only be accessed by administrative software tools within the TRE Project's virtual machine. If an encryption key file is destroyed, it is near impossible to decrypt the data volume that the key file was protecting. This method irrecoverably destroys the entire data volume. Therefore, destruction of the encryption key file is the most effective method of rendering all a TRE Project's data inaccessible by any means. An applicable scenario is when a TRE Project ends or the agreed data retention time has expired and the Data Controller requires all project data to be permanently destroyed.

The data volume destruction techniques prescribed by this SOP renders the data irretrievable by any protocol, including and not limited to:

- access to the virtual workstation that was previously mounted to the storage volume containing that datasets
- direct access to the TRE's storage array management console
- using sufficiently strong encryption cypher mechanisms such that the likelihood (and therefore risk) of a malicious actor gaining access to encrypted data by deliberately re-generating a deleted encryption key is acceptably low
- physically removing the hard disk and performing a low-level sector scan in an attempt to identify and rebuild the file-system

Permanent destruction of a data volume is a two-step procedure:

1. Destroy the storage volume header
2. Destroy the volume's encryption key-file

This method is applicable to storage volumes hosted in the TRE's main storage array appliance, local machines disks and portable storage devices.

4.3. Procedure for the Destruction of the Storage Volume Header

The TRE uses dm-crypt (LUKS) to encrypt storage volumes. This tool places the encryption key slots in the first portion of blocks in the storage volume (aka the volume 'header'). These slots typically only reside within the first 2MB of the volume. This portion of the storage volume can therefore be wiped by becoming overwritten with random characters. Overwriting the first 10MB of the storage volume ensures the encryption slots are destroyed. This process supports RAID arrays. The command for wiping the first 10MB of a disk header with random characters is:

```
# dd if=/dev/zero of=/dev/sdXY bs=1M count=10
```

4.4. Technique for the Destruction of Encryption Key Files

As an encryption key-file is simply a small file stored in a specific area of the TRE, the standard method for file destruction can be used here (see next section).

4.5. File Destruction

The UNIX *shred* command will be used to destroy files. For example, to shred a file call 'secret.txt', issue the following command:

```
# shred –uvz secret.txt
```

To destroy all files within a directory, issue the following command:

```
# find <directory> -depth -type f -exec shred -v -n 1 -z -u {} \;
```

4.6. Evidence of Data Destruction

Each time a file or data volume is destroyed using the above techniques, the shell output must be sent to a text file. This file must be exported from the TRE and stored within the ISMS (e.g. attached to the data asset record on Q-Pulse) to evidence that destruction was carried out. For example to shred the file 'secret.txt' and output the results to a file called 'file-shred-results.txt', issue the following command:

```
# nohup shred –uvz secret.txt >> file-shred-results.txt &
```

Evidence of destroying the storage volume header should also be output in a similar fashion.

Any other correspondence, e.g. email record related to the data destruction request must be stored similarly.

5. Cross-referenced ISMS Documents

Number	Type	Title

SOP-05-03	ISMS\SOP\Asset and Supplier Management - SOP	Importing Content into the TRE
SOP-03-24	ISMS\SOP\TRE Operations - SOP	Migrating Projects out of the TRE

6. Appendices

None