

# De-Identification Rules

## Public Elements

1. DICOM PS 3.15 Annex E publishes a table that encodes de-identification actions for a subset of DICOM public elements. The TagSniffer software has an implementation of this table used to generate reports that includes the hexadecimal tag of the public element, the coded action and potentially a list of all unique values found in this element in the collection or sub-collection.
2. DICOM PS 3.15 Annex E only documents elements that might reasonably contain PHI. We have extended the table used by TagSniffer to include a larger set of public elements annotated with the same coding scheme. Not all public elements have been added to the TagSniffer table. As we run into more datasets with elements that are not in our table, we can extend the table.
3. Examine the output of the TagSniffer software for elements that have clear actions that do not require review. Examples are the coded actions Z (zero), X (delete), K (keep). Encode the proper action in the CTP de-identification script.
4. Examine the TagSniffer output for elements that will require review. Some of these are encoded with R (review); others are encoded with C (clean). Note that R is a coded value defined by MIR and not DICOM PS 3.15 Annex E. Determine the proper action for each element and implement in the CTP de-identification script.
5. Exceptions:
  - a. 0008 103E (Series Description) is coded by PS 3.15 Annex E with X for delete. The NCI wants us to retain this value. Review this for PHI as if it was encoded as C or R. We need to update our implementation tables. [Comment by kes: TCIA implements the Clean Description Option which says to Clean series description, so I believe this is not an exception]
  - b. The original combination of CTP + scripts in the field entered some elements in images with zero length when the logic should have been "if present, replace with zero-length element". These require further review for DICOM SR objects. For DICOM images, delete these elements:
    - i. 0008 0100 Code Value
    - ii. 0008 0102 Coding Scheme Designator
    - iii. 0008 0104 Code Meaning
    - iv. 0040 2016 Placer Order Number
    - v. 0040 2017 Filler Order Number
    - vi. 0040 A075 Verifying Observer Name
    - vii. 0040 A088 Verifying Observer Code Sequence
    - viii. 0040 A123 Person Name
    - ix. 0070 0084 Content Creator's Name
6. These elements might be found in images but either contain information that will have PHI or are dates that are not really helpful for researchers. In the context of these dates, it is not worth the time/effort to offset the dates and then check them later to make sure they are correct. It is better to just delete them. We will get these actions encoded in our TagSniffer software:
  - a. 0032 1000 Scheduled Study Start Date
  - b. 0032 1001 Scheduled Study Start Time
  - c. 0032 1050 Study Completion Date
  - d. 0032 1051 Study Completion Time
  - e. 0040 0244 Performed Procedure Step Start Date
  - f. 0040 0245 Performed Procedure Step Start Time
  - g. 0040 0250 Performed Procedure Step End Date
  - h. 0040 0251 Performed Procedure Step End Time
  - i. 0040 0275 Request Attributes Sequence

## Private Elements

1. The TagSniffer software has profiles for modalities from vendors. The profiles are created by reading DICOM conformance statements and translating the vendor information on private elements into a table format that uses the coded vocabulary from PS 3.15 Annex E. That coded vocabulary is extended
  - a. R Review the element for PHI
  - b. KB Keep the data; according to the conformance statement, it contains binary values that do not need review.
2. Not all private elements are documented in conformance statements. A conformance statement may document 30 private elements and omit 5 that are in the images. We may not be able to locate the appropriate conformance statement. The TagSniffer output indicates the coded action as "null". You can translate that to review.
3. There will be collisions in the private element tags that can be resolved by examining the Private Creator data that goes with the private elements. The collisions are usually caused by a PACS or workstation adding extra elements after the modality has created the original data.
4. Examine the TagSniffer output for private elements. You need to be aware of collisions in private groups, elements that are known in the knowledge base and elements that are not known. In general, you want to augment the CTP script to manage these items:
  - a. Private data entered by a workstation or PACS are generally not of interest to researchers. The exception will be for objects that contain measurements. For run of the mill images, the private elements are likely related to interpretation/workflow. We want to remove these private elements.
  - b. Private elements documented in our knowledge base are presented with the same coded actions as public elements. For the elements that are designated K or KB, write the CTP script to keep these. For those labelled R, review the set of values in the TagSniffer output for PHI.
  - c. Private elements not represented in the TagSniffer profile files will be listed in TagSniffer output with a "null" action, indicating unknown. Review the set of values for PHI to determine if the private element should be retained or deleted.
5. Do not modify the values of any private elements. After review, the CTP script should either retain the value or delete the element.
6. Some of the output for the set of values in private elements is a little confusing. The dcm4che software does the best it can with retrieving private elements as strings, but there are some quirks. If you see things that are confusing, check with Steve to see if the TagSniffer software needs adjustment.

## DCIODVFY: Legal DICOM Objects

1. David Clunie did us a great service by finding images that were not legal DICOM objects. We have his software, and we now run it on one instance from each series. That output is included for both intake and public.
2. If time were infinite and engineers were free, I would recommend that we examine this output on both intake and public to make sure that we have legal objects.
3. However, in the real world.
  - a. It is possible that a site will ship us objects that are not legal DICOM objects.
  - b. It is more likely that our CTP script taking the collection from intake to public introduced some anomaly, therefore
  - c. I would recommend that we only examine the dciodvfy output when TagSniffer is run on the data in public (before release). If we find illegal objects, we can refer to the dciodfy output at the intake stage and resolve where the inconsistencies were introduced.