

Classification Marking Macros for L^AT_EX *

Keith F. Prussing
kprussing74@gmail.com

January 3, 2026

Contents

1	Introduction	2
2	Usage	2
2.1	Marking Management	2
2.2	Document Marking	3
2.3	Overriding the Default Page Marking	5
2.4	Additional Statements	5
3	Tips and Tricks	5
3.1	Do Not Span Sections	6
3.2	Do Not Wrap Floats	6
3.3	Marking Front Matter Titles	6
3.4	Marking List of Acronyms from <code>Acro</code>	7
3.5	Adjusting Boxed Environments	7
4	Implementation	7
4.1	Required Packages	7
4.2	Package Options	8
4.3	Package Counters	8
4.4	Marking and Heading Management	9
4.5	Low Level Macros	10
4.6	Marking Utilities	10
4.7	Boxed Environment Marking	11
4.8	Specific Markings	11
4.9	Overriding the Default Page Marking	13
4.10	Deprecated Macros	13
4.11	Formatting Details	14
4.12	Additional Statements	15
A	Examples	16
A.1	Example Document	16
A.2	Beamer Slides	18

*This document corresponds to `secretx v0.10.9`, dated 2024/03/07.

B	References	19
C	Change History	19
D	Index	20

Abstract

This package provides a set of L^AT_EX macros for marking paragraphs in a document with a leading string and modifying the header and the footer of each page. This is the spiritual successor to `secret.sty` but adds the feature of marking the header and the footer correctly.

1 Introduction

A core component of reporting sensitive information is properly marking sections of a document with appropriate labels. This package aims to provide tools for helping mark a document correctly.

2 Usage

To install the package, simply run the provided `secretx.ins` file through your preferred L^AT_EX engine. Then, copy the resultant `.sty` file to a location where L^AT_EX can find it. To generate the documentation, run this file (`secretx.dtx`) through L^AT_EX twice. The second time is only to make sure the references and index are typeset correctly. If you have LuaL^AT_EX, you could also run `l3build install` to install into your local texmf tree and `l3build doc` to get a quick version of the documentation.

Section [A](#) provides an example document and beamer presentation showing the usage of the package. The raw source is also provided in the `doc` directory with this reference. To see the typeset result, simply run the example files through L^AT_EX.

2.1 Marking Management

The basic concept behind marking each page is setting a “level” for each paragraph. The highest level paragraph on any given page sets the overall level of the page in a standard document. Before use, the user must specify every desired marking level along with paragraph and header markings. The header and footer markings are set with a `fancyhdr` `\pagestyle` which is activated by default for non-beamer documents.

To apply headers and footers to beamer slides, use

```
\usebeamertemplate{headline}[secretx]
\usebeamertemplate{footline}[secretx]
```

or simply use `\getheading` in your custom headline and footline. Additionally, the headline and footline themes accept an optional argument. If this matches the string “title”, the overall document banner is typeset instead of the individual page banner.

`\setmarking` To register a paragraph marking, use `\setmarking{<level>}{<marking>}`. Like `\setheading` wise, to register a heading use `\setheading{<level>}{<heading>}`. These macros

register a `{<marking>}` or `{<heading>}` at the specified `{<level>}`. The user is responsible for making sure the marker and heading for given level are consistent. Providing a level more than once will simply overwrite the previous definition with no warning.

`\addmarking` `\addmarking{<level>}{<marking>}{<heading>}` will set the marking and heading at the given level. This is simply a convenience wrapper around `\setmarking` and `\setheading`.

`\getmarking` Once a level has been registered, retrieving the marking or heading is done with `\getmarking[<level>]` or `\getheading[<level>]` respectively. The argument defaults to the current paragraph or page level as appropriate. These macros check that the given `[<level>]` has a defined marking and heading. If not, these will raise an error. These return the expanded string given to `\setmarking` and `\setheading`.

`\pagelevel` These provide access to the current page and document levels. These are read only and are provided to reveal the current state for the user instead of having to work with the package internals.

`\pagestyle{secretx}` This `pagestyle` places the appropriate header and footer page marking in the center of the header and footer. It changes no other parts of the header or footer. This is loaded by default; however, it is ultimately the users responsibility to ensure this is loaded at the appropriate time or that other header/footer customizations do not overwrite the center portion of either the header or footer. If you need to have a highly customized header or footer, simply add `\getheading` to the appropriate portion of your header or footer.

2.2 Document Marking

`\marking` Marking a paragraph is done by specifying the numeric level with which to mark using `\marking{<level>}{...}`. This macro takes care of marking the paragraph with the appropriate marker and updating the page and document levels. The page and document levels are only updated if the given marking is higher than the current level.

`marking block (env.)` Collecting large bodies of text into a macro is not ideal. Instead, we would like to use an environment. To this end, this package provides the `marking block` environment. The syntax is analogous to the `\marking` macro in that it takes a `{<level>}`. For example:

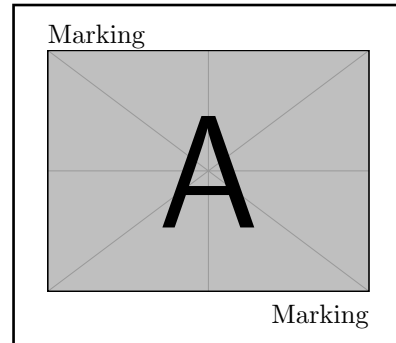
```
\begin{marking block}{1}
\ldots
\end{marking block}
```

`marking box (env.)` Graphical or tabular content can be wrapped in a `tcolorbox` to indicate the marking level of the entire content. The `marking box` environment will wrap the item in a colored box and place the heading in the upper left and lower right corners of the box in the same color as the box. The environment accepts the following options:

1. `{<level>}`: The level of the marking
2. `<{top sep}>`: The additional separation above the content
3. `<{bottom sep}>`: The additional separation below the content
4. `[<options>]`: Additional options passed to `tcolorbox`

Under the hood, this is a `tcolorbox` customized with the `secretx` style.

```
\setheading{1}{Marking}
\begin{marking box}{1}
  \includegraphics%
    [width=\textwidth]%
    {example-image-a}
\end{marking box}
```



For convenience, we provide definitions to set common levels. The styles and colors listed are used with `marking box` and may be customized as described in the documentation for `tcolorbox`. These are:

<code>\unclassified</code>	Mark unclassified.
<code>unclassified block (env.)</code>	
<code>unclassified box (env.)</code>	<ul style="list-style-type: none">• Level: 1
<code>unclassified style</code>	<ul style="list-style-type: none">• Marking: (U)
<code>unclassified/color</code>	<ul style="list-style-type: none">• Heading: Unclassified
<code>\fouo</code>	Mark for official use only.
<code>fouo block (env.)</code>	

Notice

The FOUO marking is deprecated. This will be removed in version 0.11.

- Level: 2
- Marking: (U//FOUO)
- Heading: Unclassified//For Official Use Only

<code>\confidential</code>	Mark confidential.
<code>confidential block (env.)</code>	
<code>confidential box (env.)</code>	<ul style="list-style-type: none">• Level: 5
<code>confidential style</code>	<ul style="list-style-type: none">• Marking: (C)
<code>confidential/color</code>	<ul style="list-style-type: none">• Heading: Confidential

<code>\secret</code>	Mark secret.
<code>secret block (env.)</code>	
<code>secret box (env.)</code>	<ul style="list-style-type: none">• Level: 10
<code>secret style</code>	<ul style="list-style-type: none">• Marking: (S)
<code>secret/color</code>	<ul style="list-style-type: none">• Heading: Secret

<code>\secretnoform</code>	Mark secret no foreign.
<code>secretnoform block (env.)</code>	
<code>secretnoform box (env.)</code>	<ul style="list-style-type: none">• Level: 15
<code>secretnoform style</code>	
<code>secretnoform/color</code>	

	<ul style="list-style-type: none"> • Marking: (S//NF) • Heading: Secret//No Form
<code>\topsecret</code>	Mark top secret.
<code>topsecret block (env.)</code>	<ul style="list-style-type: none"> • Level: 25
<code>topsecret box (env.)</code>	
<code>topsecret style</code>	<ul style="list-style-type: none"> • Marking: (TS)
<code>topsecret/color</code>	<ul style="list-style-type: none"> • Heading: Top Secret

2.3 Overriding the Default Page Marking

Notice

Version 0.11 will change the meaning of this option as the internal logic is still buggy. After 0.11, `heading` will *enable* the adaptive marking.

`heading` Although this package attempts to automate away many of the underlying details of marking a document, it cannot always succeed. Since it is the final responsibility of the document author to ensure appropriate overall markings, the `heading` argument to the package provides a manual override to the package determined page marking per the alternate marking advice in [1]. This option can be given alone or assigned as a string `heading=<text>`. If a string is given, the text is typeset in the header and footer verbatim. If the text is not given, the highest header marking of the document will be used for the header and footer. The header and footers will be the same on all pages in the document.

If possible, we would appreciate any representative examples of usages that fails to get marked correctly reported to the [issue tracker](#). However, due to the intended use of this package, we understand if this is not practical.

`\setlevel` This macro explicitly sets the level of a page without marking a particular item. The focus of this is more for a class like `beamer` where the slide may be composed solely of images that are themselves marked but are unknown to L^AT_EX, or floating environment such as `figure` or `table`. This lets the user explicitly state the classification of the proper page or scope.

2.4 Additional Statements

`\distribution` A distribution statement is a common need for documentation. Using `\distribution {<level>}{<reason>}{<date of determination>}{<controlling DoD office>}` will place the appropriate distribution statement following DoDI 5230.24[2]. Here `{<level>}` is the letter version of the statement (A, B, C, D, etc.).

`\exportcontrol` This macro places the export control warning in a framed box at the current location in accordance with DoDI 5230.24[2]. The default is to typeset the box at 90% of the `\textwidth`. Using `\exportcontrol[<width>]` typesets the control statement in a box of the given `[<width>]`.

3 Tips and Tricks

A few items must be kept in mind while using this package. Specifically, this package makes use of `everyhook` to manipulate the `\everypar` hook. As noted

online¹, L^AT_EX makes extensive use of `\everypar` in many unexpected places. From a practical point of view, the following sections point out best practices.

3.1 Do Not Span Sections

A marking environment should not span a sectioning command. If the document contains:

```
\begin{unclassified block}
\lipsum[1]

\chapter{The Next Chapter}

\lipsum[2]
\end{unclassified block}
```

the marking will be placed in the margin next to the chapter and not associated with the chapter text. Instead, use

```
\unclassified{\lipsum[1]}

\chapter{\getmarking[1]\setlevel{1} The Next Chapter}

\unclassified{\lipsum[2]}
```

This will set the level, correctly associate the marking with the chapter title, and place it in the table of contents. To silence the warning from `hyperref`, place the macros in the T_EX portion of `\texorpdfstring`.

3.2 Do Not Wrap Floats

Similar to sectioning commands, placing floating environments within a marking block has negative consequences. When the floating environment is placed, it will have the marking in the margin. Instead, explicitly set the level within the floating environment and get the marking for the caption.

```
\end{unclassified block}

\begin{figure}
  \begin{secret box}
    \includegraphics{example-image-16x10}
  \end{secret box}
  \caption{\getmarking[1] Different level caption.}
\end{figure}

\begin{unclassified block}
```

3.3 Marking Front Matter Titles

Markings can be added to the table of contents, list of figures, and list of tables titles using `\preto` from `etoolbox` (which is loaded by this package).

¹<https://tex.stackexchange.com/q/263744/61112>

```

\AtBeginDocument{%
  \preto\contentsname{\setlevel{1}\getmarking[1] }
  \preto\listfigurename{\setlevel{1}\getmarking[1] }
  \preto\listtablename{\setlevel{1}\getmarking[1] }
}

```

3.4 Marking List of Acronyms from Acro

The list of acronyms can be placed in a marking environment by redefining the template according to example 8² from the acro documentation.

```

\RenewAcroTemplate[list]{description}{%
  \acroheading
  \acropreamble
  \begin{unclassified block}
  \begin{description}
    \acronymsmapF{
      \item[%
        \acrowrite{short}
      ]
      \acrowrite{list}%
    }
    {\item \AcroRerun}
  \end{description}
  \end{unclassified block}
}

\prinacronyms[name={\getmarking[1] \acrotranslate{list-name}}]

```

3.5 Adjusting Boxed Environments

As mentioned, the `marking box` environment relies on `tcolorbox` with the custom `secretx` style. You may change the defaults using `\tcbset`. The following sets all marked box titles to be bold and to use a dark green to mark unclassified boxed content.

```

\tcbset{secretx/.append style={fonttitle=\bfseries},
  unclassified/color/.default=green!50!black,
}

```

4 Implementation

The basic logic is to define markings with increasing levels of sensitivity. We track these along with paired line markings and page markings in internal arrays.

4.1 Required Packages

For this package, we will need to modify the shipout, work with counters, control sequences, complex labels, and we will, obviously, need to modify the headers and footers. We add all of the packages in one location to keep things organized.

²acro.example.texsx-505891

```

1 \RequirePackage{atbegshi}
2 \RequirePackage{etoolbox}
3 \RequirePackage[excludeor]{everyhook}
4 \RequirePackage{environ}
5 \@ifclassloaded{beamer}{}{%
6   \RequirePackage{fancyhdr}%
7 }
8 \RequirePackage{kvoptions}
9 \RequirePackage{zref-user}
10 \RequirePackage{zref-abspage}
11 \@ifpackageloaded{tcolorbox}{}{%
12   \RequirePackage{tcolorbox}
13 }

```

Tcolorbox moved the `\NewTColorBox` macro from the `xparse` library in version 5.1.0. We need to include it for older versions.

```

14 \@ifpackagelater{tcolorbox}{2022/06/22}{}{\tcbuselibrary{xparse}}

```

4.2 Package Options

`\secretx@heading` This optional key-value pair is used to induce a manual override of page-level headings. All pages will be marked with the indicated heading if this option is specified.

```

15 \DeclareStringOption{heading}[default]
16 \ProcessKeyvalOptions*

```

4.3 Package Counters

Define the package counters and toggle.

`secretx@pagelevel` A counter to store the marking level of the highest current page.

```

17 \newcounter{secretx@pagelevel}

```

`secretx@doclevel` A counter to store the highest marking level of the current document. We also write out the document marking level to the auxiliary file for use on the second pass when creating the cover page.

```

18 \newcounter{secretx@doclevel}
19 \AtEndDocument{%
20   \immediate\write\@auxout{%
21     \string\setcounter{secretx@doclevel}{\thesecretx@doclevel}%
22   }
23 }

```

`secretx@currentlevel` A counter to maintain the marking level of the current scope.

```

24 \newcounter{secretx@currentlevel}

```

`secretx@markcount` A counter for the classification environment.

```

25 \newcounter{secretx@markcount}

```

4.4 Marking and Heading Management

These are the user facing macros for establishing the markings and headings. These automate away the tasks of storing the information at increasing levels and validating the levels when getting a marking or heading from the internal storage. When setting the marking or heading, we simply overwrite any previous value. It is up to the user to ensure the markings and headings are set at consistent levels. For this, we also provide a utility macro for setting both simultaneously.

`\setmarking` Set a marking level by defining the level and a line marking (e.g. (U) for unclassified, (S) for secret, etc.). This will overwrite a previously defined marking at the same level.

```
26 \newrobustcmd{\setmarking}[2]{\csdef{secretx@marking#1}{#2}}
```

`\getmarking` Get a line marking from the marking storage.

```
27 \newrobustcmd{\getmarking}[1][\thesecretx@currentlevel]{%
28   \ifcsdef{secretx@marking#1}{%
29     \csuse{secretx@marking#1}%
30   }{%
31     \PackageWarning{secretx}{%
32       Marking ##1 level was not previously defined. Is it a typo
33       or did you simply forget to define it?%
34     }
35   }
36 }
```

`\setheading` Set the heading level by defining the level and the heading (e.g. UNCLASSIFIED, SECRET, etc.). This will overwrite a previously defined marking at the given level.

```
37 \newrobustcmd{\setheading}[2]{\csdef{secretx@heading#1}{#2}}
```

`\getheading` Get the heading on the page. Use the current page level unless the `heading` option was specified. If it was, always use it.

```
38 \NewDocumentCommand{\getheading}{ o }{%
39   \IfValueTF{#1}{%
40     \ifcsdef{secretx@heading#1}{%
41       \csuse{secretx@heading#1}%
42     }{%
43       \PackageError{secretx}{Undefined heading level}{%
44         Marking ##1 level was not previously defined. Is it a
45         typo or did you simply forget to define it?%
46       }%
47     }%
48   }{%
49     \ifdefempty{\secretx@heading}{%
50       \csuse{secretx@heading\thesecretx@pagelevel}%
51     }{%
52       \ifdefstring{\secretx@heading}{default}{%
53         \csuse{secretx@heading\doclevel}%
54       }{%
55         \csuse{secretx@heading}%
56       }%
57     }
58   }%
59 }
```

`\addmarking` A helper macro to set a marking's level, section marking, and header marking.

```
60 \newrobustcmd{\addmarking}[3]{%
61   \setmarking{#1}{#2}%
62   \setheading{#1}{#3}%
63 }
```

`\pagelevel` User facing macros to get the current page and document marking level.

```
\doclevel 64 \newrobustcmd{\pagelevel}[0]{\thesecretx@pagelevel}
65 \newrobustcmd{\doclevel}[0]{\thesecretx@doclevel}
```

4.5 Low Level Macros

Some low level macros for managing the marking level state.

`\secretx@updatelevel` A macro to update the page and document levels. We only update if the given level is higher than the current level. We do this for both the page and the document.

```
66 \newrobustcmd{\secretx@updatelevel}[1]{%
67   \ifnum#1 > \thesecretx@pagelevel%
68     \setcounter{secretx@pagelevel}{#1}%
69   \fi%
70   \ifnum#1 > \thesecretx@doclevel%
71     \setcounter{secretx@doclevel}{#1}%
72   \fi%
73 }
```

4.6 Marking Utilities

We provide a macro for marking the document. These will handle the lower level details of getting the line markings and adjusting the details to set the headings. Each one starts by incrementing the marking count to track the count of the current environment, creates a label, sets the marking for each paragraph, and inserts the content. At the end, it sets an ending label.

`\marking` The workhorse macro to actually set the headings and paragraph markings. We begin by incrementing the current marking count and storing it for later use.

```
74 \newrobustcmd{\marking}[2]{%
75   \stepcounter{secretx@markcount}%
76   \edef\curcnt{\thesecretx@markcount}%
```

Next, in order to correctly reset the level in the case of a nested marking, we need to wrap everything up in a group and use `\aftergroup`. But, it appears like we need to wrap the counter setting in a macro.

```
77   \edef\reset{%
78     \setcounter{secretx@currentlevel}{\thesecretx@currentlevel}%
79   }%
80   \bgroup%
81   \aftergroup\reset%
```

Now step the environment counter to track the current environment.

```
82   \zlabel{secretx@mark-begin-\curcnt}%
83   \secretx@updatelevel{#1}%
84   \setcounter{secretx@currentlevel}{#1}%
85   #2%
```

We finish by marking the ending and releasing the lock.

```
86   \egroup%
87   \zlabel{secretx@mark-end-\currcnt}%
88 }
```

`marking block` (*env.*) We also provide an environment for marking large bodies of text with a common marking.

```
89 \NewEnviron{marking block}[1]{\marking{#1}{\BODY}}
```

4.7 Boxed Environment Marking

Define a style for creating a color box with level markings in the upper left end lower right.

```
90 \tcbset{%
91   secretx/.style n args={3}{%
92     % Basic shape
93     sharp corners,%
94     boxrule=1pt,%
95     boxsep=1pt,%
96     % The heading
97     title={\getheading[#1]},%
98     % Place the heading in the upper left and lower right
99     detach title,%
100    before upper={\tcbtitle\hfill\null\[#2]},%
101    after upper={\leavevmode\[#3]\null\hfill\tcbtitle},%
102    % Make the background transparent
103    standard jigsaw,%
104    opacityback=0.0,%
105    % Set the default color of the heading and the frame
106    secretx/color,%
107  },
108  secretx/.prefix code={\setlevel{#1}},
109  % Set the default color of the heading and the frame
110  secretx/color/.default=black,%
111  secretx/color/.style={/tcb/colframe=#1,,
112                        /tcb/coltitle=#1},
113 }
```

`marking box` Define a new color box that passes the appropriate options to `secretx` style.

```
114 \NewTColorBox{marking box}{m D<>{Opt} D<>{Opt} !O{} }{%
115   secretx={#1}{#2}{#3},%
116   % Additional options
117   #4%
118 }
```

4.8 Specific Markings

Now we define specific markings for the known classification levels. For each, we define an environment and a macro.

An empty default marking.

```
119 \addmarking{0}{}{}
```

```

\unclassified Unclassified markings.
unclassified block (env.) 120 \addmarking{1}{(U)}{Unclassified}
121 \newrobustcmd{\unclassified}[1]{\marking{1}{#1}}
122 \NewEnviron{unclassified block}{\unclassified{\BODY}}
123 \tcbset{unclassified/.style 2 args={secretx={1}{#1}{#2},
124                                     unclassified/color},
125         unclassified/color/.forward to=/tcb/secretx/color}
126 \NewTColorBox{unclassified box}{D<>{Opt} D<>{Opt} !0{}}
127         }{unclassified=#1}{#2}, #3}

\fouo For official use only markings.
fouo block (env.) 128 \addmarking{2}{(U//FOUO)}{Unclassified//For Official Use Only}
129 \newrobustcmd{\fouo}[1]{%
130     \PackageWarning{secretx}{%
131         Deprecation warning: Marking FOUO is not correct.\MessageBreak%
132         Define and use a CUI level instead.\MessageBreak%
133         This level will be removed in v0.11.\MessageBreak%
134     }%
135     \marking{2}{#1}%
136 }
137 \NewEnviron{fouo block}{\fouo{\BODY}}

\confidential Confidential markings.
confidential block (env.) 138 \addmarking{5}{(C)}{Confidential}
139 \newrobustcmd{\confidential}[1]{\marking{5}{#1}}
140 \NewEnviron{confidential block}{\confidential{\BODY}}
141 \tcbset{confidential/.style 2 args={secretx={5}{#1}{#2},
142                                     confidential/color},
143         confidential/color/.forward to=/tcb/secretx/color}
144 \NewTColorBox{confidential box}{D<>{Opt} D<>{Opt} !0{}}
145         }{confidential=#1}{#2}, #3}

\secret Secret markings.
secret block (env.) 146 \addmarking{10}{(S)}{Secret}
147 \newrobustcmd{\secret}[1]{\marking{10}{#1}}
148 \NewEnviron{secret block}{\secret{\BODY}}
149 \tcbset{secret/.style 2 args={secretx={10}{#1}{#2},
150                                 confidential/color},
151         secret/color/.forward to=/tcb/secretx/color}
152 \NewTColorBox{secret box}{D<>{Opt} D<>{Opt} !0{}}
153         }{secret=#1}{#2}, #3}

\secretnoform Secret no form markings.
secretnoform block (env.) 154 \addmarking{15}{(S//NF)}{Secret//No Forn}
155 \newrobustcmd{\secretnoform}[1]{\marking{15}{#1}}
156 \NewEnviron{secretnoform block}{\secretnoform{\BODY}}
157 \tcbset{secretnoform/.style 2 args={secretx={15}{#1}{#2},
158                                     secretnoform/color},
159         secretnoform/color/.forward to=/tcb/secretx/color}
160 \NewTColorBox{secretnoform box}{D<>{Opt} D<>{Opt} !0{}}
161         }{secretnoform=#1}{#2}, #3}

\topsecret Top secret markings.
topsecret block (env.)

```

```

162 \addmarking{25}{(TS)}{Top Secret}
163 \newrobustcmd{\topsecret}[1]{\marking{25}{#1}}
164 \NewEnviron{topsecret block}{\topsecret{\BODY}}
165 \tcbset{topsecret/.style 2 args={secretx={25}{#1}{#2},
166                               topsecret/color},
167         topsecret/color/.forward to=/tcb/secretx/color}
168 \NewTColorBox{topsecret box}{D<>{0pt} D<>{0pt} !0{}}
169                ){topsecret={#1}{#2}, #3}

```

4.9 Overriding the Default Page Marking

`\setlevel` This macro provides a user interface to explicitly set the page level. It is targeted as beamer frames that are predominantly explicitly marked.

```

170 \newrobustcmd{\setlevel}[1]{%
171   \secretx@updatelevel{#1}%
172 }

```

4.10 Deprecated Macros

Notice

These macros will be removed in v0.11.

These macros are deprecated and are preserved for temporary backwards compatibility. They will be removed in the future.

```

173 \newrobustcmd{\UCL}[1]{%
174   \PackageWarning{secretx}{Macro deprecated! Use
175   \protect\unclassified\space instead}%
176   \unclassified{#1}%
177 }
178 \newrobustcmd{\CNF}[1]{%
179   \PackageWarning{secretx}{Macro deprecated! Use
180   \protect\confidential\space instead}%
181   \confidential{#1}%
182 }
183 \newrobustcmd{\SEC}[1]{%
184   \PackageWarning{secretx}{Macro deprecated! Use
185   \protect\secret\space instead}%
186   \secret{#1}%
187 }
188 \newrobustcmd{\SNF}[1]{%
189   \PackageWarning{secretx}{Macro deprecated! Use
190   \protect\secretnoform\space instead}%
191   \secretnoform{#1}%
192 }
193 \newrobustcmd{\TS}[1]{%
194   \PackageWarning{secretx}{Macro deprecated! Use
195   \protect\topsecret\space instead}%
196   \topsecret{#1}%
197 }

```

4.11 Formatting Details

Now we move on to specific formatting details. The header and footer are determined by the marking level of the current page, but only for article style documents and not beamer slides. For beamer, we need to provide a headline and footline template. We can use our macros to set them once. We only specify the center header and footer and leave the left and right marks up to the user.

For beamer, simply use

```
\usebeamertemplate{headline}[secretx]
\usebeamertemplate{footline}[secretx]
```

to get a centered marking on the header and footer. To customize your headline and footline, simply add `\getheading` to your template. See section [A.2](#) for a simple example.

```
198 \@ifclassloaded{beamer}{%
199   \defbeamertemplate*{headline}{secretx}[1] []{%
200     \begin{beamercolorbox}[ht=5ex]{headline}
201       \usebeamerfont{secretx}\hfill%
202       \ifstrequal{#1}{title}{\getheading[\doclevel]}{\getheading}%
203       \hfill\null%
204     \end{beamercolorbox}
205   }
206   \defbeamertemplate*{footline}{secretx}[1] []{%
207     \begin{beamercolorbox}[ht=6ex]{footline}
208       \usebeamerfont{secretx}\hfill%
209       \ifstrequal{#1}{title}{\getheading[\doclevel]}{\getheading}%
210       \hfill\null\vspace{3pt}%
211     \end{beamercolorbox}
212   }
213   \setbeamerfont{secretx}{size=\Large,%
214     series=\bfseries,%
215     parent=structure}
216 }{%
217   \fancypagestyle{secretx}{%
218     \fancyhead[C]{\getheading}
219     \fancyfoot[C]{\getheading}
220   }
221   \pagestyle{secretx}
222 }
```

We also place the marking before every paragraph based on the current marking level.

```
223 \PushPreHook{par}{%
224   \ifcsemt{secretx@marking\thesecretx@currentlevel}%
225   {}{\setbox0=\lastbox\llap{\getmarking{}}%
226     \hspace{5pt}%
227     \hspace{\secretx@listmargin}}%
228   \box0}%
229 }
```

And now we need to deal with resetting the page levels correctly. Each marking has been placed between begin and end labels. We can check the last label when a page ships out to see if the begin is on the same page. If it is not, we need to

preserve the page level on the next page. Otherwise, we need to reset the page to empty.

```

230 \AtBeginShipout{%
231   \ifnum\zref@extract{secretx@mark-begin-\thesecretx@markcount}{abspage}
232     =\zref@extract{secretx@mark-end-\thesecretx@markcount}{abspage}
233     \setcounter{secretx@pagelevel}{0}%
234   \else%
235     \setcounter{secretx@pagelevel}{\thesecretx@currentlevel}%
236   \fi%
237 }

```

`secretx@listlevel` This counter is used to determine the nesting of the list level. It will be used to determine if the current paragraph is within a noted list environment to return the correct padding. We hook into each known list environment³.

```

238 \newcounter{secretx@listlevel}
239 \setcounter{secretx@listlevel}{0}
240 \AtBeginEnvironment{itemize}{\addtocounter{secretx@listlevel}{1}}
241 \AtEndEnvironment{itemize}{\addtocounter{secretx@listlevel}{-1}}
242 \AtBeginEnvironment{enumerate}{\addtocounter{secretx@listlevel}{1}}
243 \AtEndEnvironment{enumerate}{\addtocounter{secretx@listlevel}{-1}}
244 \AtBeginEnvironment{description}{\addtocounter{secretx@listlevel}{1}}
245 \AtEndEnvironment{description}{\addtocounter{secretx@listlevel}{-1}}

```

`secretx@listmargin` This defines an additional offset for list environments to make sure the marking does not overhang the label.

```

246 \newrobustcmd\secretx@listmargin[0]{%
247   \ifnumgreater{\value{secretx@listlevel}}{0}{\leftmargin}{0pt}
248 }

```

4.12 Additional Statements

`\distribution` Define the wrapper for directing to the proper distribution statement.

```

249 \newrobustcmd\distribution[4]{%
250   DISTRIBUTION STATEMENT #1. \cuse{secretx@distro@#1}{#2}{#3}{#4}%
251 }

```

The individual statements from Table 5 in [2].

```

252 \newrobustcmd\secretx@distro@A[3]{%
253   Approved for public release: distribution unlimited.
254 }
255 \newrobustcmd\secretx@distro@B[3]{%
256   Distribution authorized to U.S. Government agencies #1 #2. Other
257   requests for this document shall be referred to #3.
258 }
259 \newrobustcmd\secretx@distro@C[3]{%
260   Distribution authorized to U.S. Government agencies and their
261   contractors #1 #2. Other requests for this document shall be
262   referred to #3.
263 }
264 \newrobustcmd\secretx@distro@D[3]{%
265   Distribution authorized to Department of Defense and U.S. DoD

```

³This could be done with L^AT_EX3 using `AddToHook`, but that is a **recent addition**.

```

266     contractors only #1 #2. Other requests for this document shall be
267     referred to #3.
268 }
269 \newrobustcmd\secretx@distro@E[3]{%
270     Distribution authorized to DoD Components only #1 #2. Other requests
271     for this document shall be referred to #3.
272 }
273 \newrobustcmd\secretx@distro@F[3]{%
274     Further dissemination only as directed by #3 #2 or higher DoD
275     authority.
276 }

```

`\exportcontrol` Typeset the export control statement in a framed box.

```

277 \newrobustcmd\exportcontrol[1][0.9\textwidth]{
278     \fbox{%
279         \begin{minipage}{#1}%
280             WARNING - This document contains technical data whose export
281             is restricted by the Arms Export Control Act (Title 22,
282             U.S.C., Sec 2751, et seq.) or the Export Administration Act
283             of 1979 (Title 50, U.S.C., App. 2401 et seq.), as amended.
284             Violations of these export laws are subject to severe
285             criminal penalties. Disseminate in accordance with
286             provisions of DoD Directive 5230.25.
287         \end{minipage}%
288     }%
289 }

```

A Examples

This is the code demonstrating the use of `secretx`. The source is in the `doc/latex/secretx` directory. Simply run the code through L^AT_EX to see the typeset examples.

A.1 Example Document

```

% \iffalse
%<*document>
% \fi
\documentclass{article}
\usepackage{secretx}
\title{\unclassified{An example document with markings}}
\author{Keith F. Prussing}
\date{\today}

% Show the page and document level in the header.
\fancyhead[L]{Document --- \doclevel}
\fancyhead[R]{Page --- \pagelevel}
\fancyfoot[L]{\thepage}
% Remove the header line.
\renewcommand{\headrulewidth}{0pt}
% Redefine headers and markings to avoid unnecessary alarm.
\addmarking{1}{(1)}{Trivial}

```

```

\addmarking{5}{(5)}{Sensitive}
\addmarking{10}{(10)}{Important}
\addmarking{25}{(25)}{Very Important}

\usepackage{mwe}
\usepackage[colorlinks]{hyperref}
\begin{document}
\maketitle
\thispagestyle{secretx}
\tableofcontents
\listoffigures

\begin{abstract}
\unclassified{%
  This is an example document showing the usage of the
  \textsf{secretx} package. The markings in the following are purely
  for demonstration purpose. All of the material is
  unclassified.\footnote{\unclassified{%
    This document may have been typeset with the markings adjusted
    to prevent unnecessary alarm. To see the real markings, simply
    remove the redefinitions as described in
    \href{file://secretx.pdf}{secretx.pdf} and rerun through
    \LaTeX{}}.
  }}
}
\end{abstract}

\unclassified{\lipsum[1]}\par
\unclassified{\lipsum[2]}\par
\unclassified{\lipsum[3]}\par
\confidential{\lipsum[4]}\par
\unclassified{\lipsum[5]}\par
\unclassified{\lipsum[6]}\par
\topsecret{\lipsum[7]}\par
\secret{\lipsum[8]}\par
\section{\setlevel{1}\getmarking[1] A New Section}
\unclassified{\lipsum[9]\par\secret{\lipsum[9]\par}\lipsum[9]}\par
\unclassified{\lipsum[10]}\par
\unclassified{\lipsum[11]}\par
\unclassified{\lipsum[12]}\par
\unclassified{\lipsum[13]}\par
\unclassified{\lipsum[14]}\par
\unclassified{\lipsum[15]}\par

% Demonstrate marking a figure.
\begin{figure}
  \centering
  \begin{secret box}
    \includegraphics{example-image-16x10}%
  \end{secret box}
  \caption{\setlevel{1}\getmarking[1]%
    An example image with a classification on the caption.
  }

```

```

\end{figure}

% Use as an environment
\begin{unclassified block}
  \lipsum[16-20]
\end{unclassified block}

\begin{secret block}
  \section{A Section within a Block}

  Notice how the marking gets placed in the margin and not correctly
  with the section title? It is not properly reflected in the table
  of contents. Additionally, Fig. \ref{fig:bad-figure} places the
  markings in the margin and not with the image or the caption.

  \lipsum[21]\par
  \unclassified{\lipsum[22]}\par
  \lipsum[23]\par
  \begin{figure}
    \includegraphics{example-image-16x10}
    \unclassified{\caption{A short caption}}
    \label{fig:bad-figure}
  \end{figure}
\end{secret block}

\begin{itemize}
  \item \unclassified{The first unclassified item}
  \item \confidential{The second item is a higher marking}
  \item \unclassified{The third is back to unclassified}
\end{itemize}

\unclassified{\lipsum[1]\par}

\begin{enumerate}
  \item \confidential{First enumerated item}
  \item \unclassified{A second enumerated item}
  \item \unclassified{And a final enumerated item}
\end{enumerate}

\begin{description}
  \item[One] \unclassified{First description item}
  \item[Two] \unclassified{A second description item}
  \item[Three] \confidential{And a final description item}
\end{description}
\end{document}
% \iffalse
%</document>
% \fi
%
```

A.2 Beamer Slides

```
% \iffalse
```

```

%<*slides>
% \fi
\documentclass{beamer}

\usepackage{secretx}

\begin{document}
\begin{frame}
  \frametitle{\unclassified{An example}}
  \begin{itemize}
    \item \unclassified{An unclassified item}
  \end{itemize}
\end{frame}

\begin{frame}
  \frametitle{\unclassified{A slide with secret information}}
  \begin{itemize}
    \item \secret{A secret item}
  \end{itemize}
\end{frame}
\end{document}
% \iffalse
%</slides>
% \fi
%
```

B References

- [1] *Marking Classified Information Job Aid*, Jul. 2021. Accessed: Nov. 28, 2022. [Online]. Available: https://www.cdse.edu/Portals/124/Documents/jobaids/information/Marking_Classified_Information.pdf?ver=K6RVqlnVuvTy051I2yF23g%3d%3d
- [2] F. Kendall, Ed., *Distribution Statements on Technical Documents*, DoD Instruction, Department of Defense, Washington, DC, Oct. 2018. Accessed: Mar. 13, 2021. [Online]. Available: <https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/523024p.pdf>

C Change History

v0.10	control statement	16	v0.11		
General: Add distribution statements		15	General: Add marking box environment		3
Add setlevel command		5	Aim for the margin using everyhook's recommended method.		14
Allow default behavior of header		5	Allow selecting the document level for header/footer		14
Define fancyhdr page style		3	<code>secretx@listlevel</code> : Add list		
Do not overhang the margin		14			
Move development to trunk branch		1			
<code>\exportcontrol</code> : Add export					

counter for margins	15	labels	10
<code>secretx@listmargin</code> : Define		v0.8.3	
margin macro for markings lists	15	General: Use PushPostHook for	
<code>\getheading</code> : Get specific heading		<code>everypar</code>	14
when given level	9	v0.8.4	
v0.5		General: Add beamer example . .	18
General: Initial version	1	Move example article to	
v0.6		appendix	16
General: Implementation		Provide beamer head/footer . .	14
reorganization	1	Use current level as default	
v0.7		marking	3
General: Correctly roll over		v0.8.5	
marking level	1	General: Add explicit	
v0.7.1		environments	1
General: Expand usage		v0.8.6	
documentation	2	General: Make all new commands	
v0.7.2		robust	1
General: Use everyhook to mark		v0.8.7	
paragraphs	14	General: Adjust beamer	
v0.8		header/footer to keep size . . .	14
General: Consolidate macros	1	v0.9.0	
v0.8.1		General: Provide manual heading	
<code>\marking</code> : Remove nested marking		override	1
restriction	10	v0.9.1	
v0.8.2		<code>fouo_block</code> : Fix capitalization . .	12
<code>\marking</code> : Resolve multiply defined			

D Index

Symbols

`\@ifpackagelater` 14

A

`\addmarking` 3,
60, 119, 120, 128, 138, 146, 154, 162

`\addtocounter`
. 240, 241, 242, 243, 244, 245

C

`\CNF` 178

`\confidential` 4, 138, 180, 181

confidential block (env.) 4, 138

confidential box (env.) 4

`\confidential_style` 4

`\confidential/color` 4

D

`\distribution` 5, 249

`\doclevel` 3, 53, 64, 202, 209

E

environments:

 confidential block 4, 138

confidential box 4

fouo block 4, 128

marking block 3, 89

marking box 3

secret block 4, 146

secret box 4

secretnoform block 4, 154

secretnoform box 4

topsecret block 5, 162

topsecret box 5

unclassified block 4, 120

unclassified box 4

`\exportcontrol` 5, 277

F

`\fouo` 4, 128

fouo block (env.) 4, 128

G

`\getheading` 3, 38, 97, 202, 209, 218, 219

`\getmarking` 3, 27, 225

H	
<code>\heading</code>	5
<code>\hspace</code>	226, 227
I	
<code>\ifstrequal</code>	202, 209
M	
<code>\marking</code>	3,
	74, 89, 121, 135, 139, 147, 155, 163
marking block (env.)	3, 89
<code>\marking_box</code>	114
marking box (env.)	3
P	
<code>\pagelevel</code>	3, 64
<code>\pagestyle\{secretx\}</code>	3
S	
<code>\SEC</code>	183
<code>\secret</code>	4, 146, 185, 186
secret block (env.)	4, 146
secret box (env.)	4
<code>\secret_style</code>	4
<code>\secret/color</code>	4
<code>\secretnoform</code>	4, 154, 190, 191
secretnoform block (env.)	4, 154
secretnoform box (env.)	4
<code>\secretnoform_style</code>	4
<code>\secretnoform/color</code>	4
<code>\secretx@currentlevel</code>	24
<code>\secretx@distro@A</code>	252
<code>\secretx@distro@B</code>	255
<code>\secretx@distro@C</code>	259
<code>\secretx@distro@D</code>	264
<code>\secretx@distro@E</code>	269
<code>\secretx@distro@F</code>	273
<code>\secretx@doclevel</code>	18
<code>\secretx@heading</code>	15, 49, 52
<code>\secretx@listlevel</code>	238
<code>\secretx@listmargin</code> ...	227, 246, 246
<code>\secretx@markcount</code>	25
<code>\secretx@pagelevel</code>	17
<code>\secretx@updatelevel</code> ...	66, 83, 171
<code>\setheading</code>	2, 37, 62
<code>\setlevel</code>	5, 108, 170
<code>\setmarking</code>	2, 26, 61
<code>\SNF</code>	188
T	
<code>\tcbuselibrary</code>	14
<code>\topsecret</code>	5, 162, 195, 196
topsecret block (env.)	5, 162
topsecret box (env.)	5
<code>\topsecret_style</code>	5
<code>\topsecret/color</code>	5
<code>\TS</code>	193
U	
<code>\UCL</code>	173
<code>\unclassified</code>	4, 120, 175, 176
unclassified block (env.)	4, 120
unclassified box (env.)	4
<code>\unclassified_style</code>	4
<code>\unclassified/color</code>	4