


1.1 screw cap



Three large, stylized question marks are arranged horizontally. Below each question mark is a small, empty square box, likely for a response or answer.



Three large, stylized question marks are arranged horizontally. Below each question mark is a small, empty square checkbox.





1 Screw Cap -



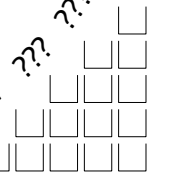
Three large question marks are arranged horizontally. Below each question mark is a small, empty square checkbox.

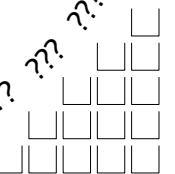
?	?	?		
?	?			
?				

Harry H.



Three large, stylized question marks are arranged horizontally. Below each question mark is a small, empty square checkbox.





Qv14
Natural, Dumpy

[illegible][illegible][illegible]

???

???

???

[illegible][illegible][illegible][illegible]

Diagram illustrating the structure of a 128-bit block cipher, showing the input, intermediate rounds, and the output.

The input is a 128-bit block, represented by three question marks (???) and a vertical line (|).

The input is processed through 12 rounds of computation, each represented by a vertical line (|).

The output is a 128-bit block, represented by five stars (☆☆☆☆☆) and a vertical line (|).

[illegible]

Times										Eye										Nose										Mouth										Score																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR	CPR

Diagram illustrating the structure of a 100-bit data stream, divided into three sections: 33 bits, 33 bits, and 34 bits.

The first section (33 bits) is labeled "???" and contains 10 vertical lines representing bits.

The second section (33 bits) contains 10 vertical lines representing bits.

The third section (34 bits) contains 10 vertical lines representing bits, followed by a star symbol (★) indicating a specific bit position.

[illegible]

The diagram illustrates a sequence of 20 data points. The first three points are marked with '???' and the last five points are marked with stars. The points are arranged in a horizontal line with vertical tick marks below them.

[illegible][illegible][illegible]

[illegible]

[illegible][illegible][illegible][illegible][illegible]

Times					Eye					Nose					Mouth					Score				
Quevedo	2014	JDAW	Natural, Straight	Quevedo	2014	JDAW	Natural, Straight	Quevedo	2014	JDAW	Natural, Straight	Quevedo	2014	JDAW	Natural, Straight	Quevedo	2014	JDAW	Natural, Straight	Quevedo	2014	JDAW	Natural, Straight	Quevedo
Qv14																								☆☆☆☆
Natural, Straight																								

Natural, Dumpy					Quevedo					2014					JDAW					Natural, Dumpy				
Quevedo	2014	JDAW	Natural, Dumpy	Quevedo	2014	JDAW	Natural, Dumpy	Quevedo	2014	JDAW	Natural, Dumpy	Quevedo	2014	JDAW	Natural, Dumpy	Quevedo	2014	JDAW	Natural, Dumpy	Quevedo	2014	JDAW	Natural, Dumpy	Quevedo
Qv14																								☆☆☆☆
Natural, Dumpy																								

Quevedo					2014					JDAW					Screw Cap					Quevedo				
Quevedo	2014	JDAW	Screw Cap	Quevedo	2014	JDAW	Screw Cap	Quevedo	2014	JDAW	Screw Cap	Quevedo	2014	JDAW	Screw Cap	Quevedo	2014	JDAW	Screw Cap	Quevedo	2014	JDAW	Screw Cap	Quevedo
Qv14																								☆☆☆☆
Screw Cap																								

Artificial					Quevedo					2014					JDAW					Artificial				
Artificial	Quevedo	2014	JDAW	Artificial	Quevedo	2014	JDAW	Artificial	Quevedo	2014	JDAW	Artificial	Quevedo	2014	JDAW	Artificial	Quevedo	2014	JDAW	Artificial	Quevedo	2014	JDAW	Artificial
Qv14																								☆☆☆☆
Artificial																								

AHB					AHB					AHB					AHB					AHB				
AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB	AHB
???																								☆☆☆☆

MPM					MPM					MPM					MPM					MPM				
MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM	MPM
???																								☆☆☆☆

[illegible][illegible]

Diagram illustrating a sequence of data points (I and H) over time. The sequence starts with three 'I' points (marked with '???' in the original image) followed by 17 'H' points (marked with stars in the original image).

[illegible][illegible][illegible]

Mike M.

[illegible][illegible][illegible][illegible][illegible][illegible]

[illegible][illegible][illegible]

Qv14
Artificial

[illegible][illegible]

[illegible][illegible][illegible][illegible][illegible]

???

???

[illegible][illegible][illegible][illegible][illegible][illegible]

Wine Of The Night?

[illegible]

The Accounts

	— Bestowals — Already paid	Wines	— Share of costs — Wines	Food etc	Owes	— Settlement — Is owed	Paid?
Julian W.							
Ian J.							
Neil C.							
Alex B.							
Tony C.							
Mike M.							
Harry H.							
Charles R.							
Total							

The Closures

The figure displays a 14x14 grid of data points, organized into four quadrants by a central cross. Each quadrant represents a different material/finish combination, with the central cross representing the year 2014. The quadrants are labeled as follows:

- Top-Left Quadrant:** Natural, Straight
- Top-Right Quadrant:** Artificial
- Bottom-Left Quadrant:** Natural, Dumpy
- Bottom-Right Quadrant:** ???

The grid is divided into four quadrants by a central cross. Each quadrant represents a different material/finish combination, with the central cross representing the year 2014. The quadrants are labeled as follows:

- Top-Left Quadrant:** Natural, Straight
- Top-Right Quadrant:** Artificial
- Bottom-Left Quadrant:** Natural, Dumpy
- Bottom-Right Quadrant:** ???

The grid is divided into four quadrants by a central cross. Each quadrant represents a different material/finish combination, with the central cross representing the year 2014. The quadrants are labeled as follows:

- Top-Left Quadrant:** Natural, Straight
- Top-Right Quadrant:** Artificial
- Bottom-Left Quadrant:** Natural, Dumpy
- Bottom-Right Quadrant:** ???

The Closures

???

The Closures

???



<https://GitHub.com/jdaw1/placemat/#readme> <http://GitHub.com/jdaw1/placemat/blob/main/PostScript/placemat.ps?raw=1>

```
:-) Execution successfully completed. George Burns: "Happiness is having a large, loving, caring, close-knit family, in another city." :-)
© Copyright 2026 Julian D. A. Wiseman of www.JDAWiseman.com
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International Licence.
http://creativecommons.org/licenses/by-sa/4.0/deed.en
product = GPL Ghostscript; languagelevel = 3; version = 3010; revision = 10060; buildtime = 0
Distilled 23:43 Tue 27 Jan 2026, according to the clock of the computer running the distillation application.
Software version = SoftwareVersionDateTimeAdobeFormat = D:202601222245 = 22:45 Thu 22 Jan 2026
These placemats ~ ParametersVersionDateTimeAdobeFormat = D:202601272345 = 23:45 Tue 27 Jan 2026
Used paper types: A3 = 297mm*420mm; A4 = 210mm*297mm.
External links for document outline:
Thread on ThePortForum.com = = http://www.ThePortForum.com/viewtopic.php?t=16619&view=unread#unread
Latest version this placemat = http://www.JDAWiseman.com/2026/20260225\_Quevedo\_closure.pdf
in list of placemats = http://www.JDAWiseman.com/papers/placemat/placemats\_list.html#a20260225
Boot & Flogger = = = http://www.davy.co.uk/wine-bar/boot-and-flogger/
...OpenStreetMap = http://www.OpenStreetMap.org/node/534961998
...google.co.uk = http://goo.gl/maps/7NEiwAMEZpG2
...What3Words: cafe.humid.palace = http://map.what3words.com/cafe.humid.palace
...StreetMap.co.uk = http://www.StreetMap.co.uk/map.srf?x=532409&y=180083&z=106&pc=SE1+1TA
...bing.com = http://www.bing.com/maps?where1=SE1+1TA
GitHub = = = = = http://GitHub.com/jdaw1/placemat/#readme
PostScript = http://GitHub.com/jdaw1/placemat/blob/main/PostScript/placemat.ps
Documentation start = http://GitHub.com/jdaw1/placemat/blob/main/Documentation/introduction\_first\_placemat.md#readme
Author = = = = = http://www.JDAWiseman.com/author.html
Fonts: /Asap-Bold (TitlesFont, BelowtitlesFont, SubtitlesFont); /Asap-Regular (CirclearraysFont, NamesFont, HeaderFont); and perhaps others.
Array equalities: {Circlearrays = CirclearraysTastingNotes = CirclearraysVoteRecorder = CirclearraysCorkDisplay}; {Titles =
TitlesTastingNotes = TitlesVoteRecorder = TitlesCorkDisplay}; {Belowtitles = SubtitlesTastingNotes = SubtitlesVoteRecorder =
SubtitlesCorkDisplay}; {Names = NamesTastingNotes = NamesVoteRecorder = NamesAccounts}.
SheetNum=0, with 13 glasses: best BaseStyle, with radius 128.918, is /Diamonds with 5 rows and 5 columns; max. capacity of this arrangement
being 13. Of non-margin area 75.4018% within circles = 83.1423% of infinite-plane exact-hexagonal maximum.
Radii = 128.918 ==> diameter ~ 91.0mm ~ 3.58".
GlassesOnSheets = [ [ 0 1 2 3 4 5 6 7 8 9 10 11 12 ] ]
Titles, ASCIIified and re-arranged by GlassesOnSheets = [ [ (Qv14) (Qv14) (Qv14) (Qv14) (???) (???) (???) (???) (???) (???) (???) (???) (???) ] ]
GlassesOnTastingNotePages = [ [ 0 1 2 3 4 5 ] [ 6 7 8 9 10 11 12 ] ]
TitlesTastingNotes, ASCIIified and re-arranged by GlassesOnTastingNotePages =
[
[ (Qv14) (Qv14) (Qv14) (Qv14) (???) (???) ]
[ (???) (???) (???) (???) (???) (???) (???) (???) ]
]
CirclearraysFontSizeMax = 12; CirclearraysFontSizes = 11.9414.
CirclearraysFontSizes optimisation, 317 steps, as AvgExcessGap - 2 * FontSize; ScoresBest = [ -1.42603 ].
CirclearraysUnroundedN = [ [ 3.53415 3.59471 4.11752 4.3528 25.0102 22.3461 26.4002 24.8978 31.6828 18.386 35.2236 32.3611 32.3611 ] ];
CirclearraysN = [ [ 3 3 4 4 25 22 26 24 31 18 32 32 32 ] ].
ProportionBaselineUsed, avg by sheet = 0.950025.
RadiiCirclearrayInside = 117.373; RadiiCirclearrayBaseline = 126.39; Radii = 128.918 (already logged).
Binding constraints on TitleFontSizes as SheetNum,WithinPage,WithinTitles,Title: 0,0,0,Qv14; 0,1,1,Qv14; 0,2,2,Qv14; 0,3,3,Qv14; 0,4,4,???;
0,5,5,???; 0,6,6,???; 0,7,7,???; 0,8,8,???; 0,9,9,???; 0,10,10,???; 0,11,11,???; 0,12,12,???.
TitleFontSizes = [ [ 104.04 104.04 104.04 104.04 166.039 166.039 166.039 166.039 166.039 166.039 166.039 166.039 ] ].
TitleFontSizes, equality classes: [ [ 0 0 0 0 1 1 1 1 1 1 1 1 ] ]; frequencies=[ 4 9 ]; #1/#0=1.59591.
TitleFontSizes, collated equality classes: 0 = Qv14 Qv14 Qv14 Qv14; 1 = ??? ??? ??? ??? ??? ??? ??? ??? ??? ????.
Title heights / RadiiCirclearrayInsideUsableTAB = [ [ 0.756057 0.756057 0.756057 0.756057 1.00316 1.00316 1.00316 1.00316 1.00316 1.00316 1.00316 ] ].
BelowtitleFontSizes = [ [ 25.3059 25.3059 25.3059 25.3059 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ] ].
BelowtitleFontSizes/TitleFontSizes = [ [ 0.243232 0.243232 0.243232 0.243232 small small small small small small small small ] ].
0,4,4,??? has max TitleAboveBelowOverT ~ 58.8718; 0,0,0,Qv14 has min TitleAboveBelowOverB ~ -80.07.
VoteRecorderSheetNum=0: VoteRecorderSubtitleFontSizeProportionTitles at least once replaced with 0.5, a font size of 11.5189pt.
NamesVoteRecorder, sorted shorter-to-longer: Ian J.; Neil C.; Alex B.; Tony C.; Mike M.; Harry H.; Julian W.; Charles R. Names with lengths
locally non-monotonic: Ian J.
URL # tags: Glasses_0 ... Glasses_8; TastingNotes_0 ... TastingNotes_17; VoteRecorder_0; Accounts_0; CorkDisplay_0 ... CorkDisplay_2;
DistillationLog; and also 13 glass-circle zooms of form Circle_#NameNum_#SheetNum_#WithinPage, from Circle_00_00_00 to Circle_00_00_12. (But
distillation might not be in Distiller, so these # tags might be missing.)
Execution time ~ 8.25 seconds, excluding time for parameter assignments, log page(s), and output.
:-) 0 = count: should = 0
:-) 1 = vmstatus pop pop: should = 1|2
:-) 0 = CountClipStack: should = 0
:-) 0 = CountGraphicsStack: should = 0
:-) 3 = countdictstack: should = 3
```