CSC 18

John Ryan Zelling Analyst

Jan 2017

Contents

CSC 18.0

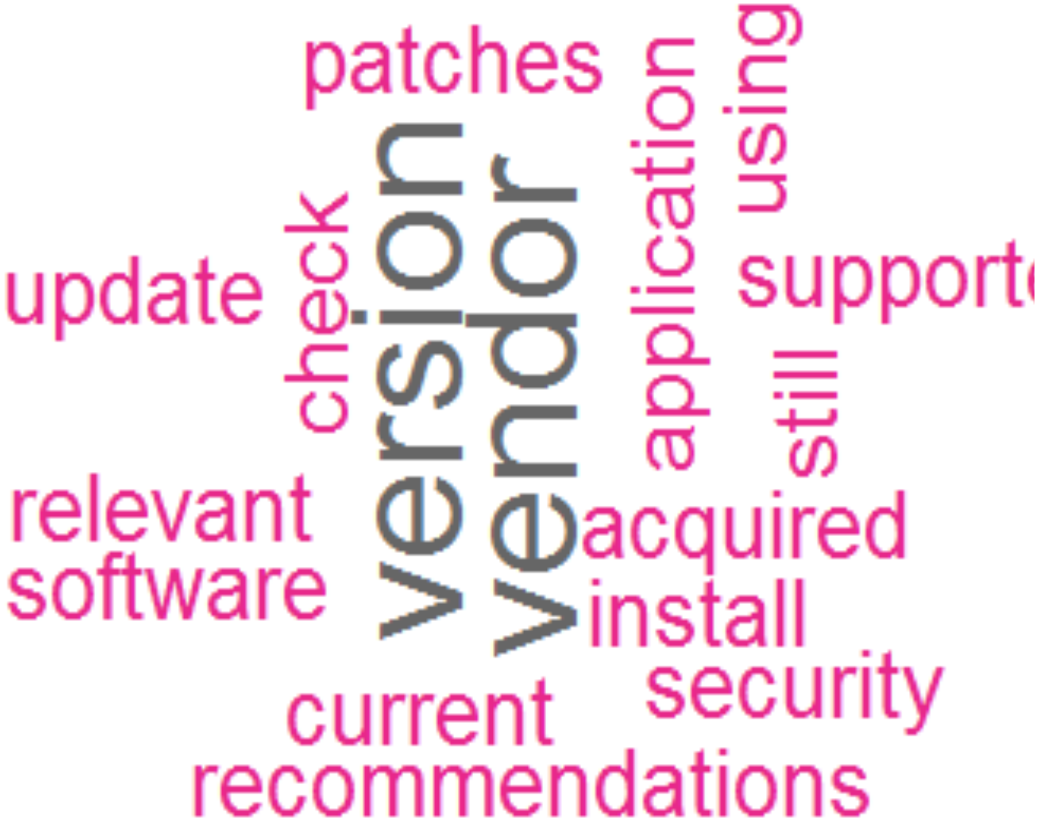
[1] "Critical Security Control #18: Application Software Security"

[1]

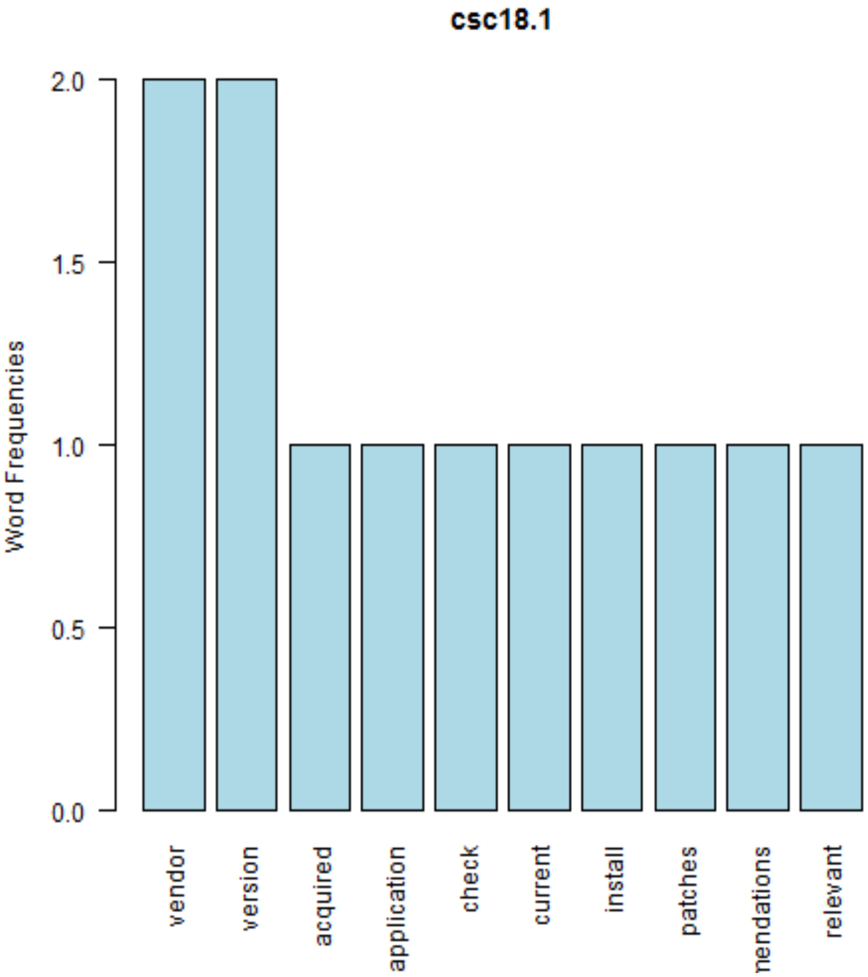
[1] [1] "To further clarify the Creative Commons license related to the CIS Critical Security Controls content, you are authorized to copy and redistribute the content as a framework for use by you, within your organization and outside of your organization for non-commercial purposes only, provided that (i) appropriate credit is given to CIS, and (ii) a link to the license is provided. Â Additionally, if you remix, transform or build upon the CIS Critical Security Controls, you may not distribute the modified materials. Users of the CIS Critical Security Controls framework are also required to refer to (http://www.cisecurity.org/critical-controls.cfm) when referring to the CIS Critical Security ControlsÂ in order to ensure that users are employing the most up to date guidance. Â Commercial use of the CIS Critical Security Controls is subject to the prior approval of The Center for Internet Security."

CSC 18.1

[1] "vendor + version"



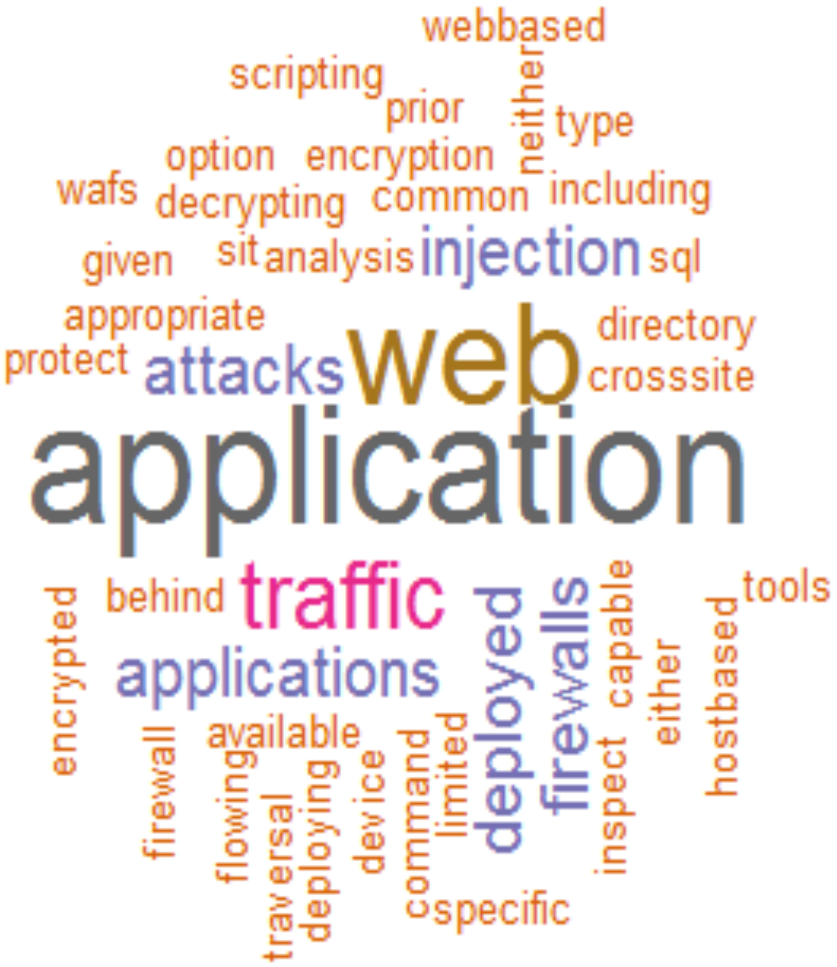
null device 1



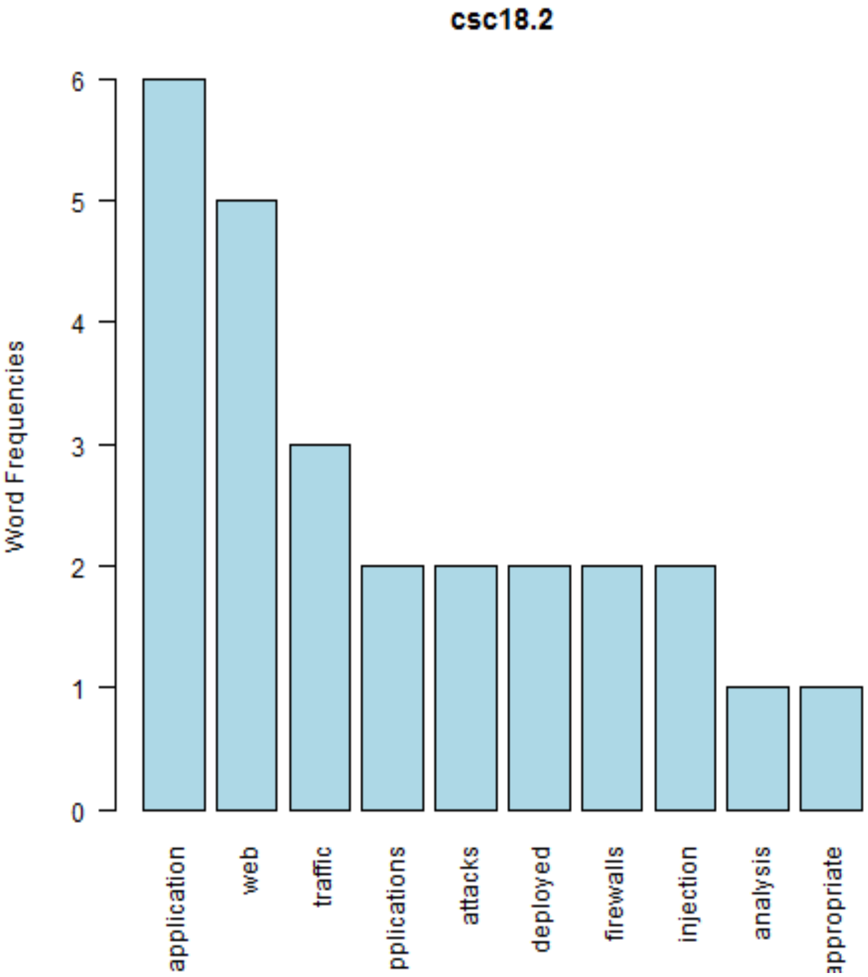
null device 1 [1] “For all acquired application software, check that the version you are using is still supported by the vendor. If not, update to the most current version and install all relevant patches and vendor security recommendations.”

CSC 18.2

[1] “application + web”



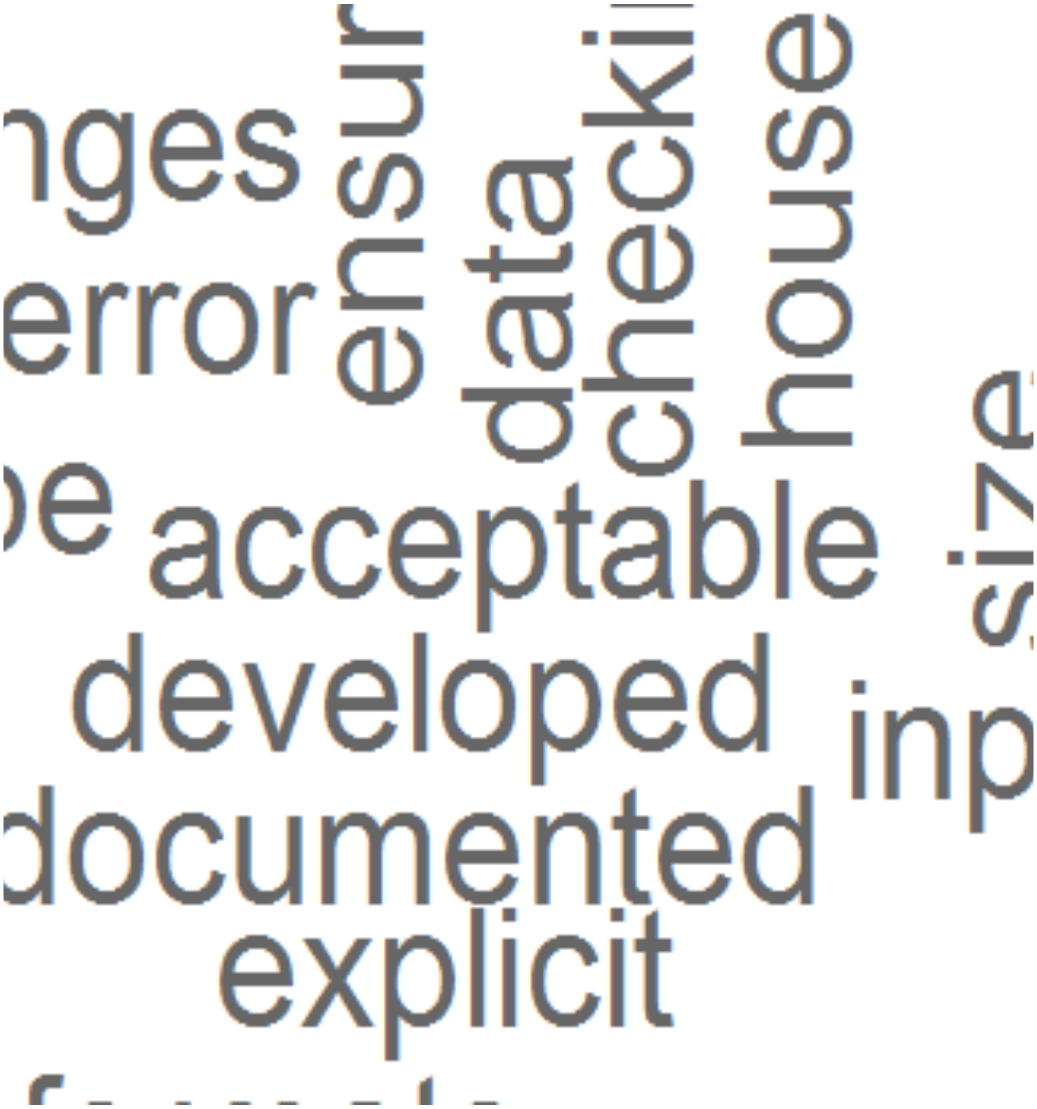
null device 1



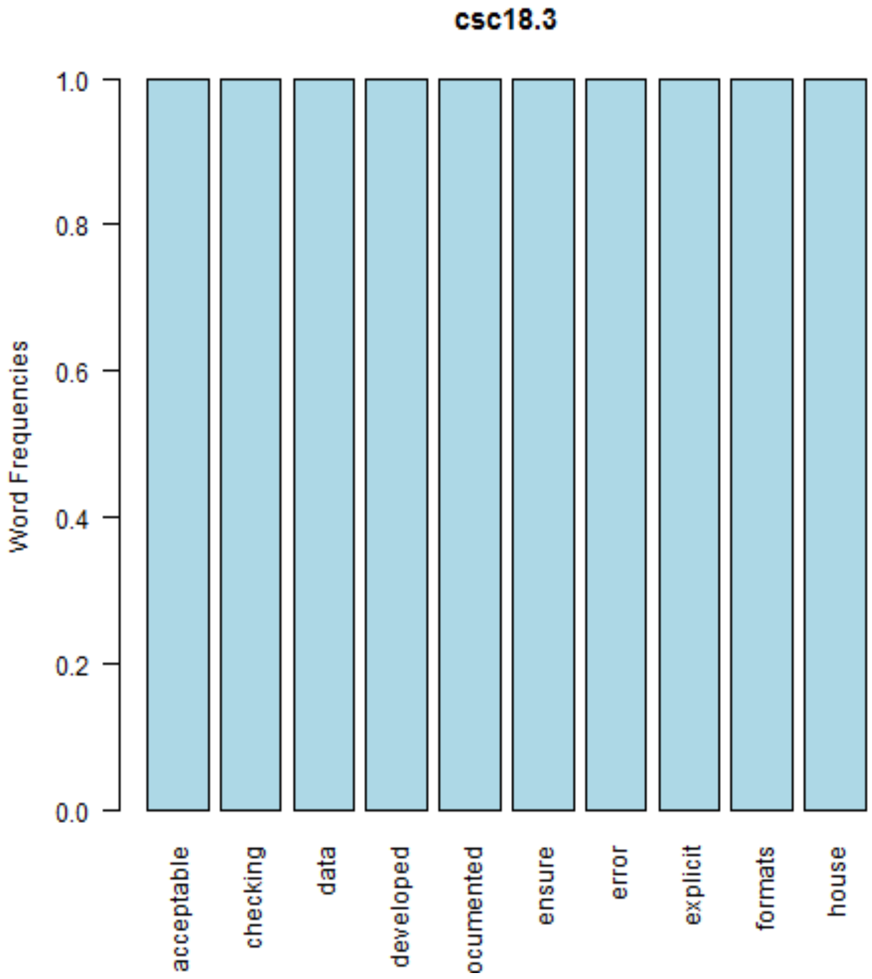
null device 1 [1] “Protect web applications by deploying web application firewalls (WAFs) that inspect all traffic flowing to the web application for common web application attacks, including but not limited to cross-site scripting, SQL injection, command injection, and directory traversal attacks. For applications that are not web-based, specific application firewalls should be deployed if such tools are available for the given application type. If the traffic is encrypted, the device should either sit behind the encryption or be capable of decrypting the traffic prior to analysis. If neither option is appropriate, a host-based web application firewall should be deployed.”

CSC 18.3

[1] “acceptable + checking”



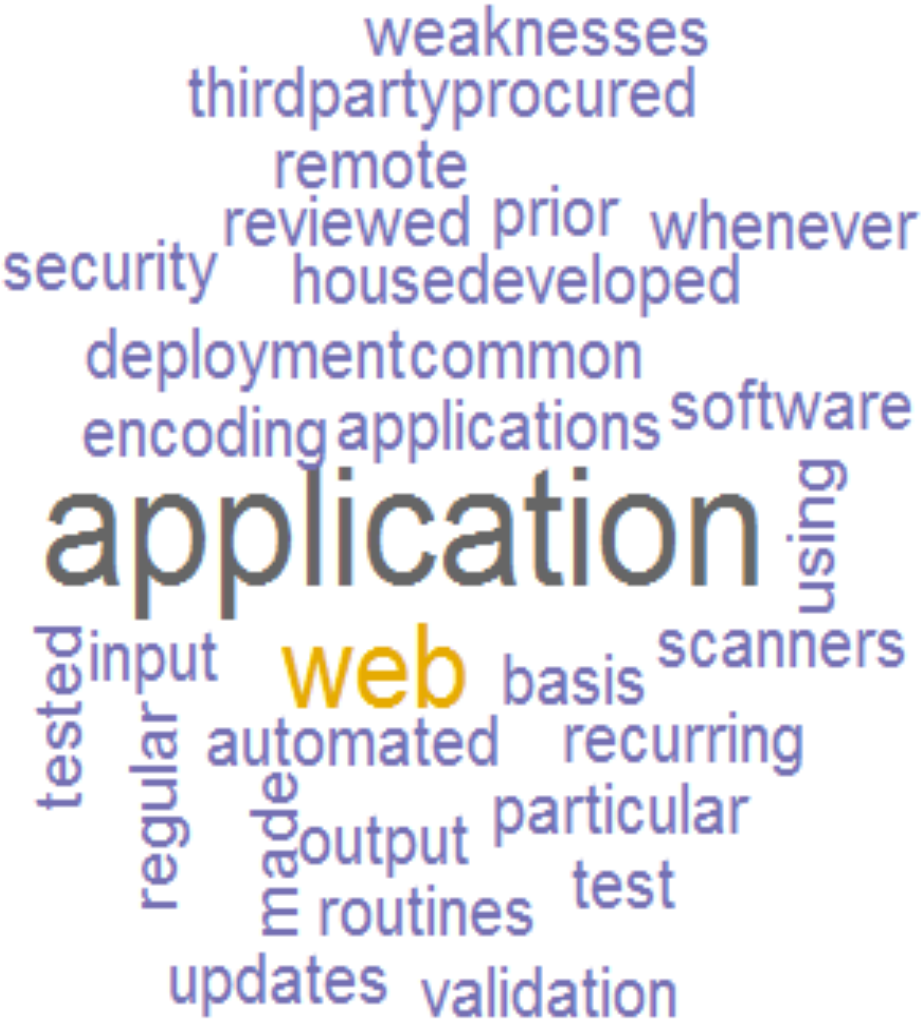
null device 1



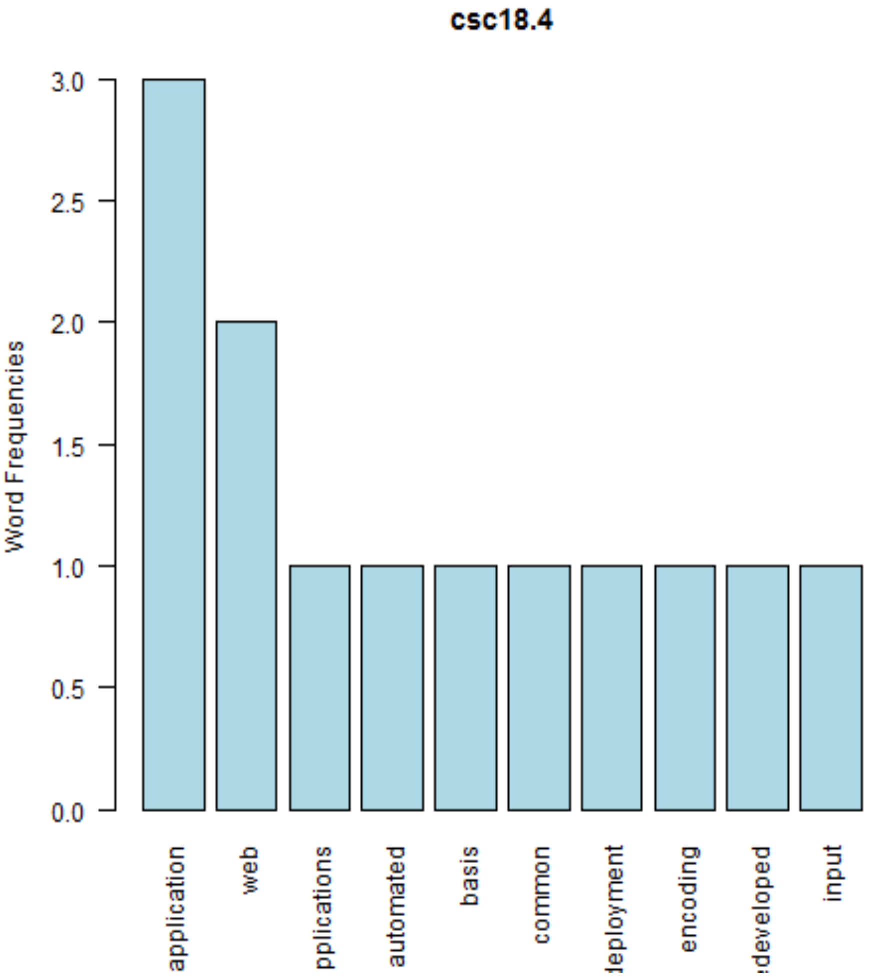
null device 1 [1] “For in-house developed software, ensure that explicit error checking is performed and documented for all input, including for size, data type, and acceptable ranges or formats.”

CSC 18.4

[1] “application + web”



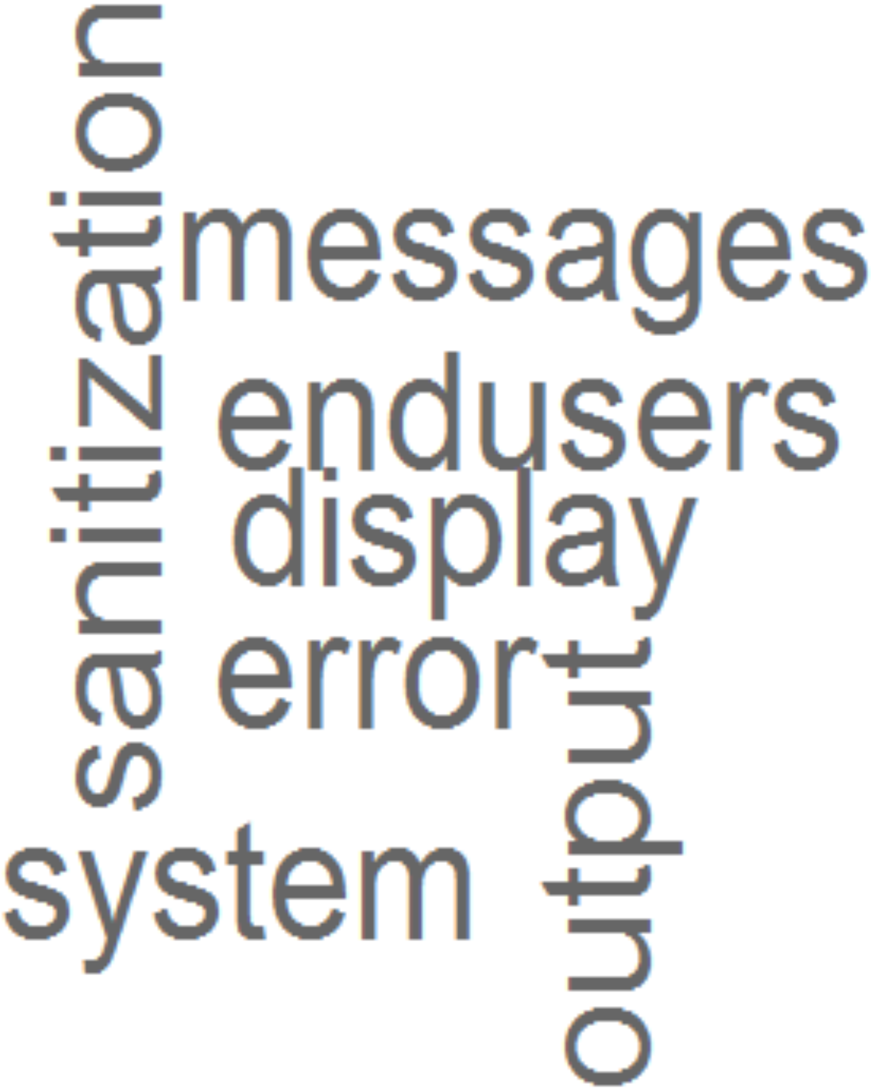
null device 1



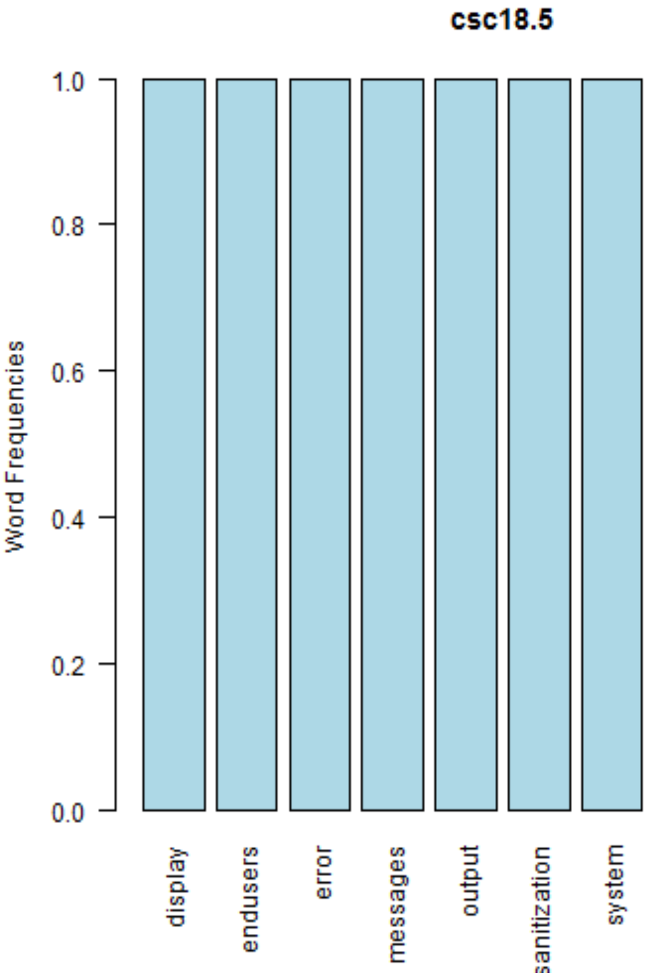
null device 1 [1] “Test in-house-developed and third-party-procured web applications for common security weaknesses using automated remote web application scanners prior to deployment, whenever updates are made to the application, and on a regular recurring basis. In particular, input validation and output encoding routines of application software should be reviewed and tested.”

CSC 18.5

[1] "display + endusers"



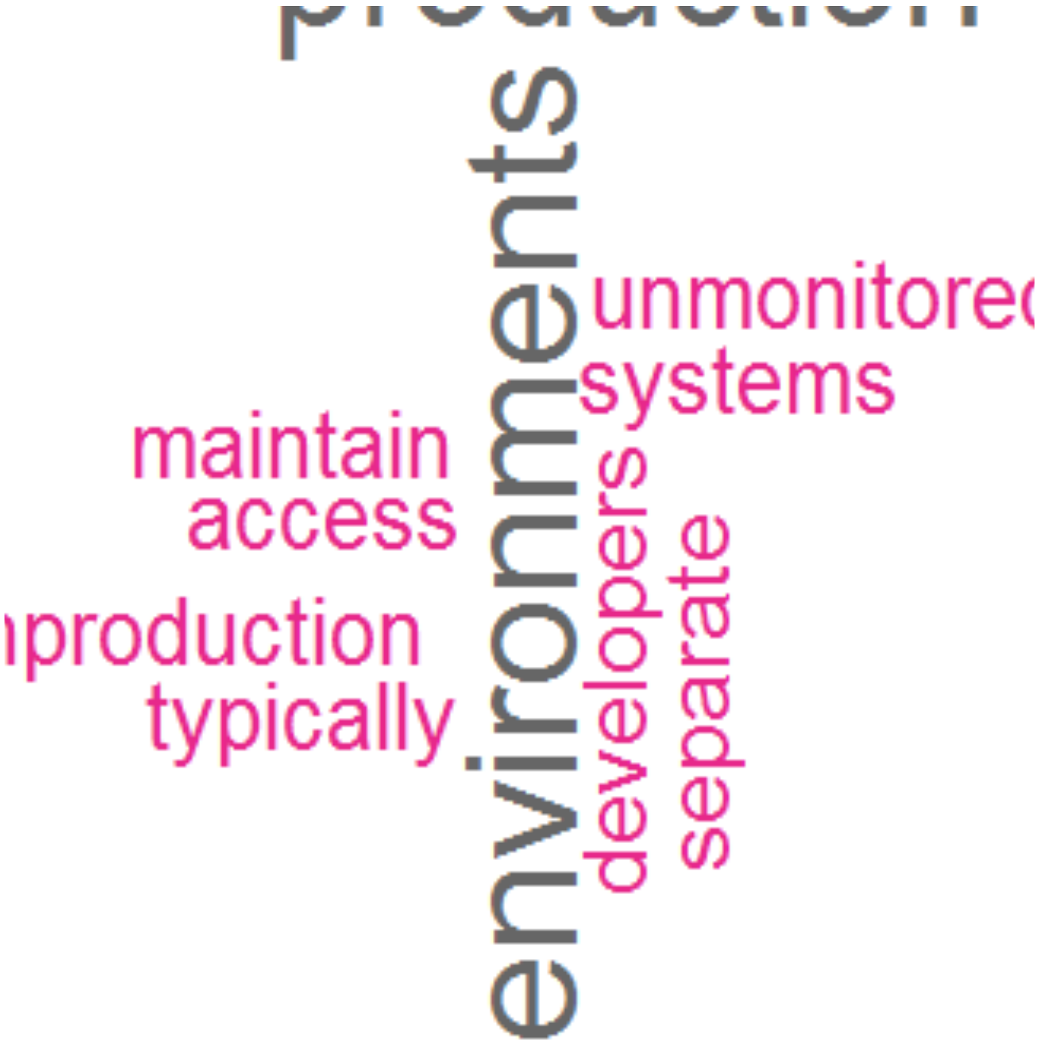
null device 1



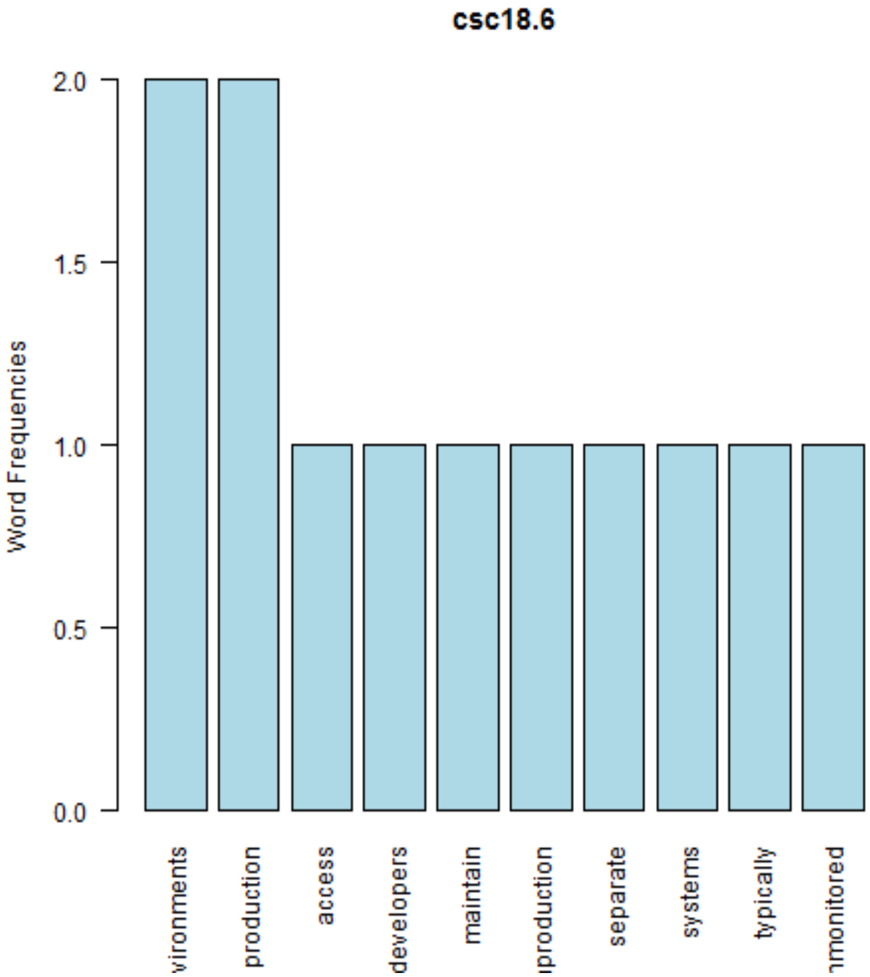
null device 1 [1] “Do not display system error messages to end-users (output sanitization).”

CSC 18.6

[1] “environments + production”



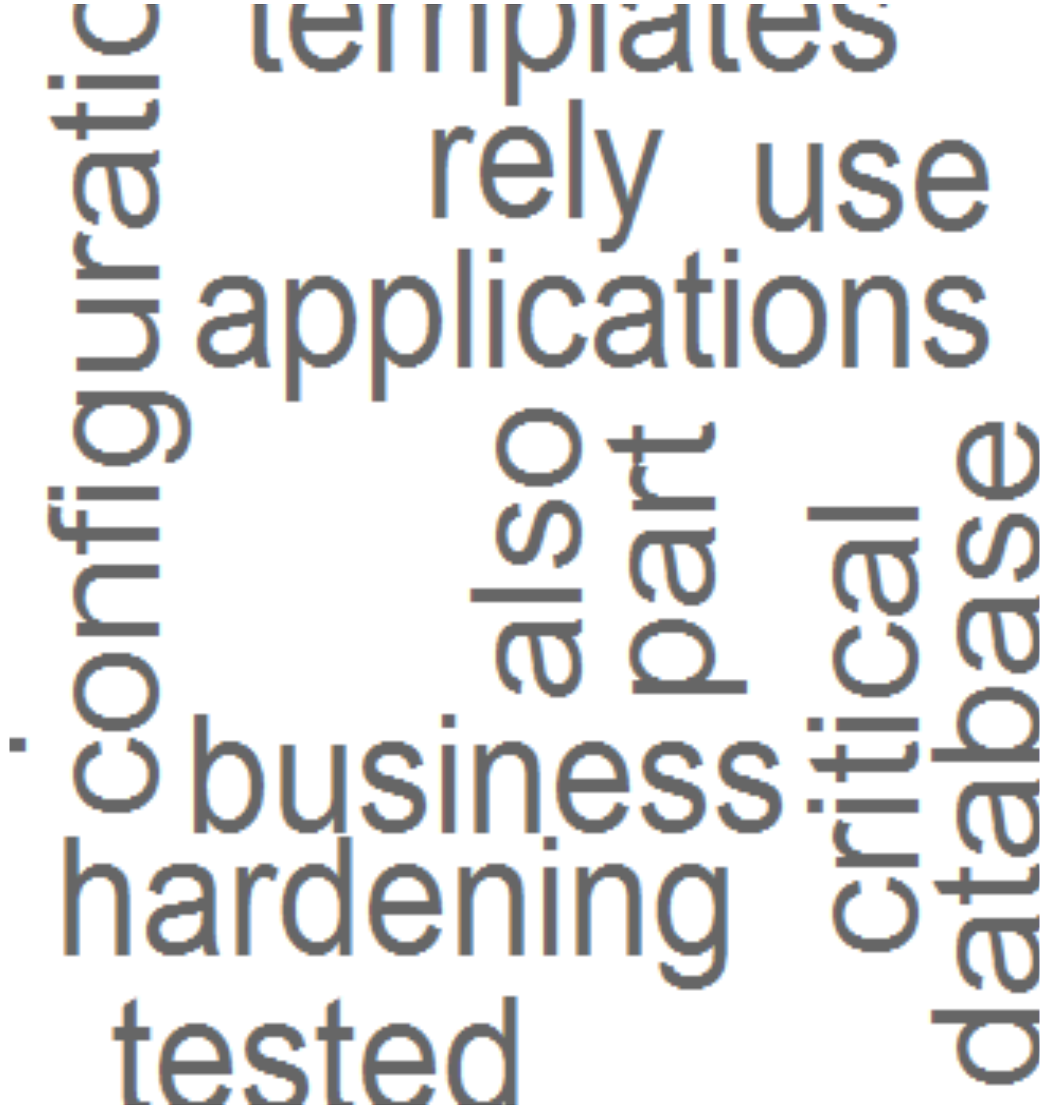
null device 1



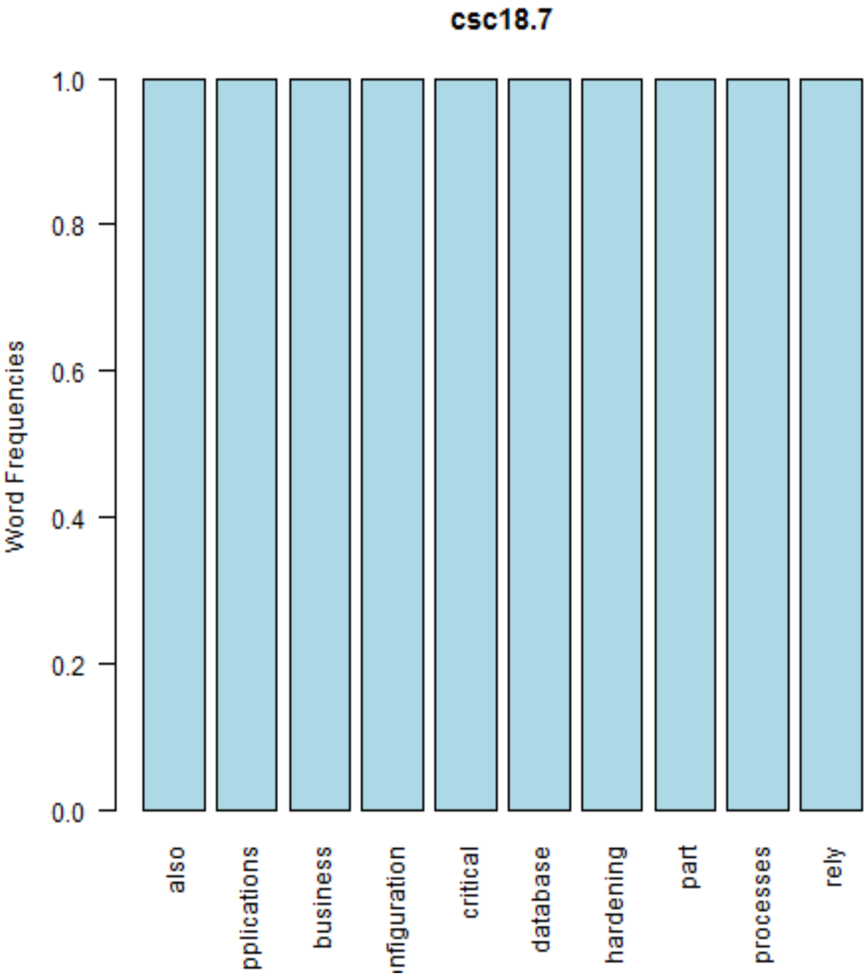
null device 1 [1] "Maintain separate environments for production and nonproduction systems. Developers should not typically have unmonitored access to production environments."

CSC 18.7

[1] "also + applications"



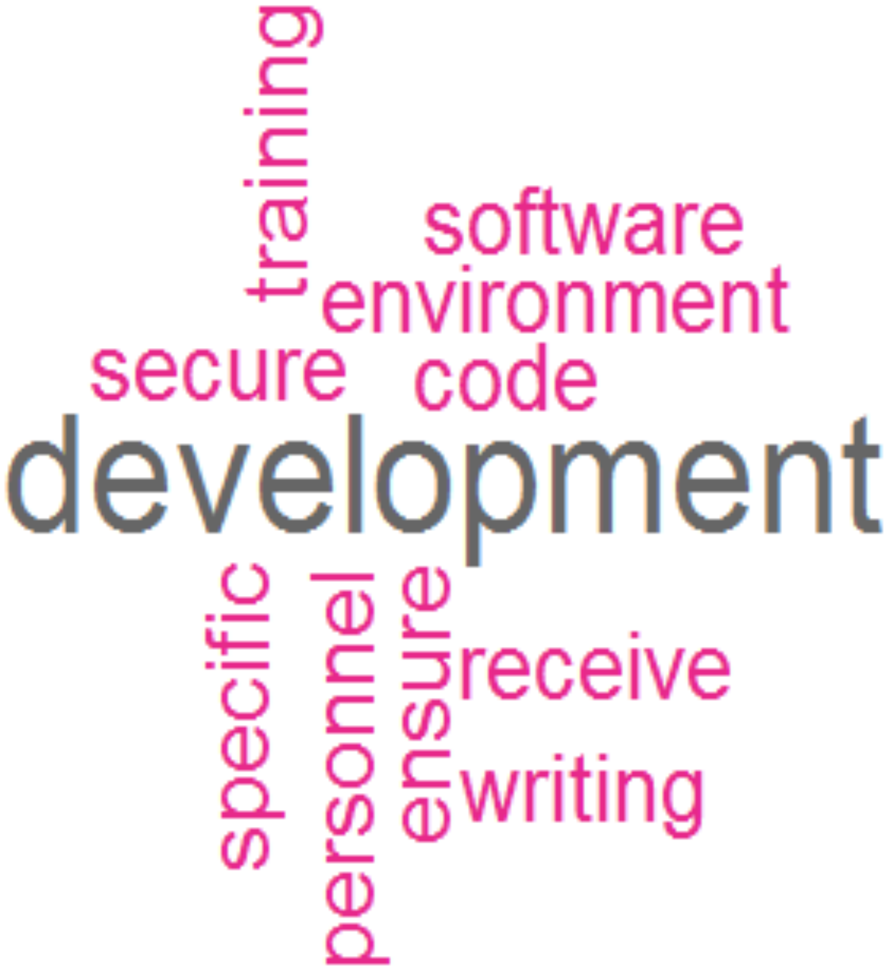
null device 1



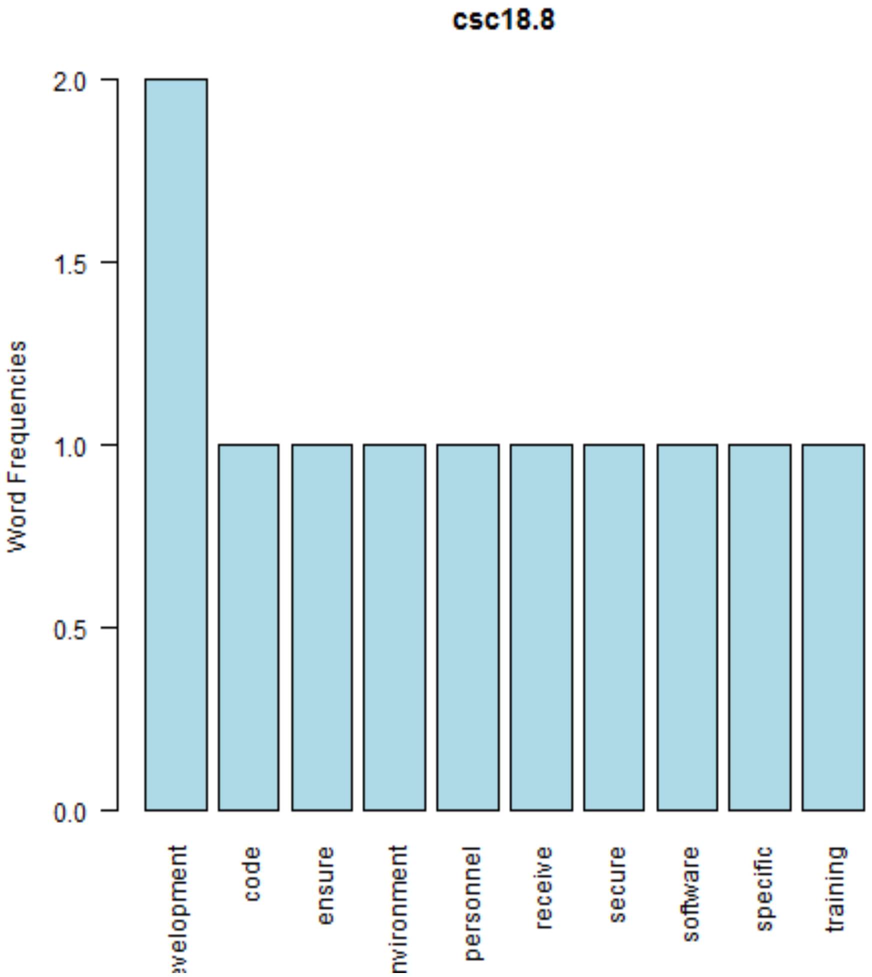
null device 1 [1] “For applications that rely on a database, use standard hardening configuration templates. All systems that are part of critical business processes should also be tested.”

CSC 18.8

[1] “development + code”



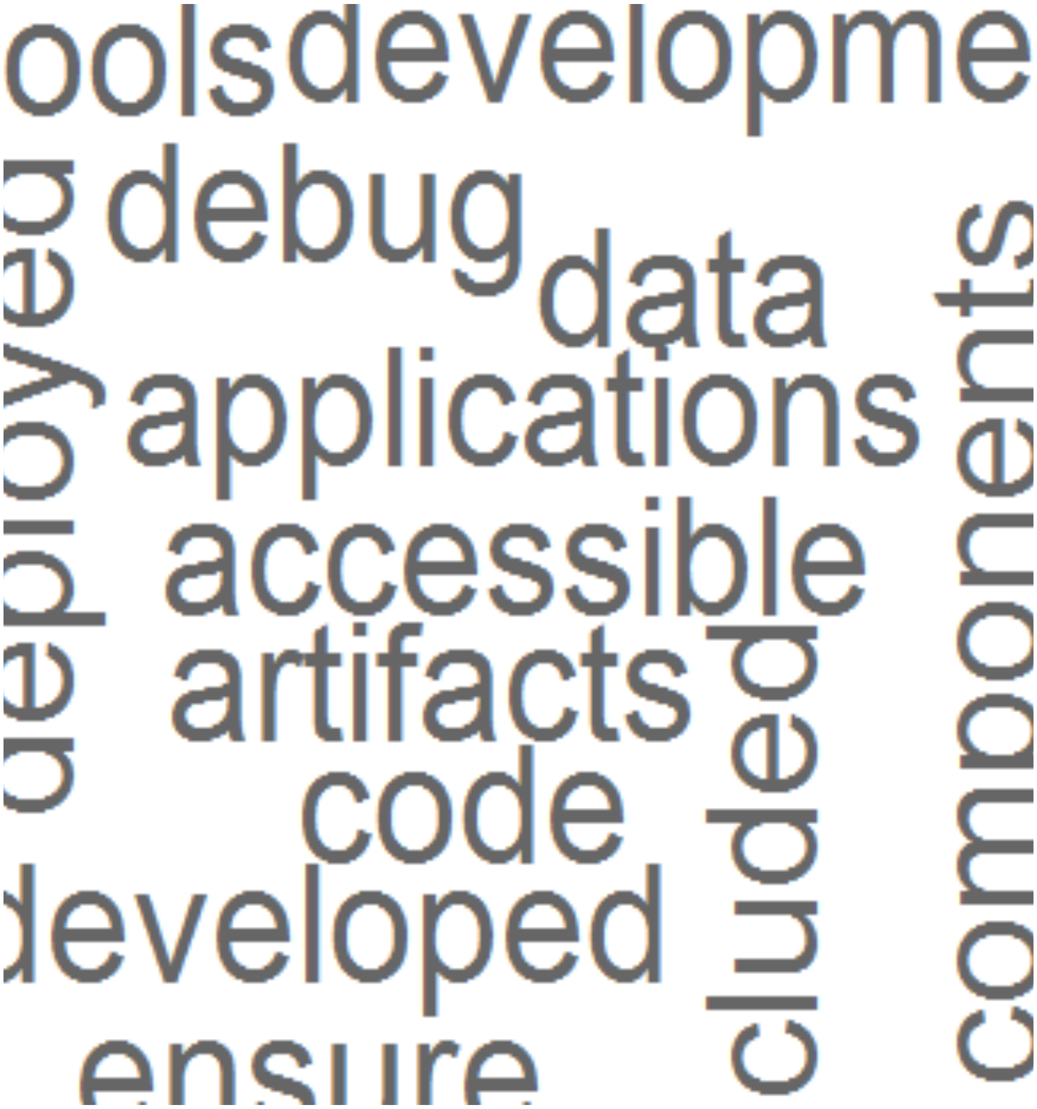
null device 1



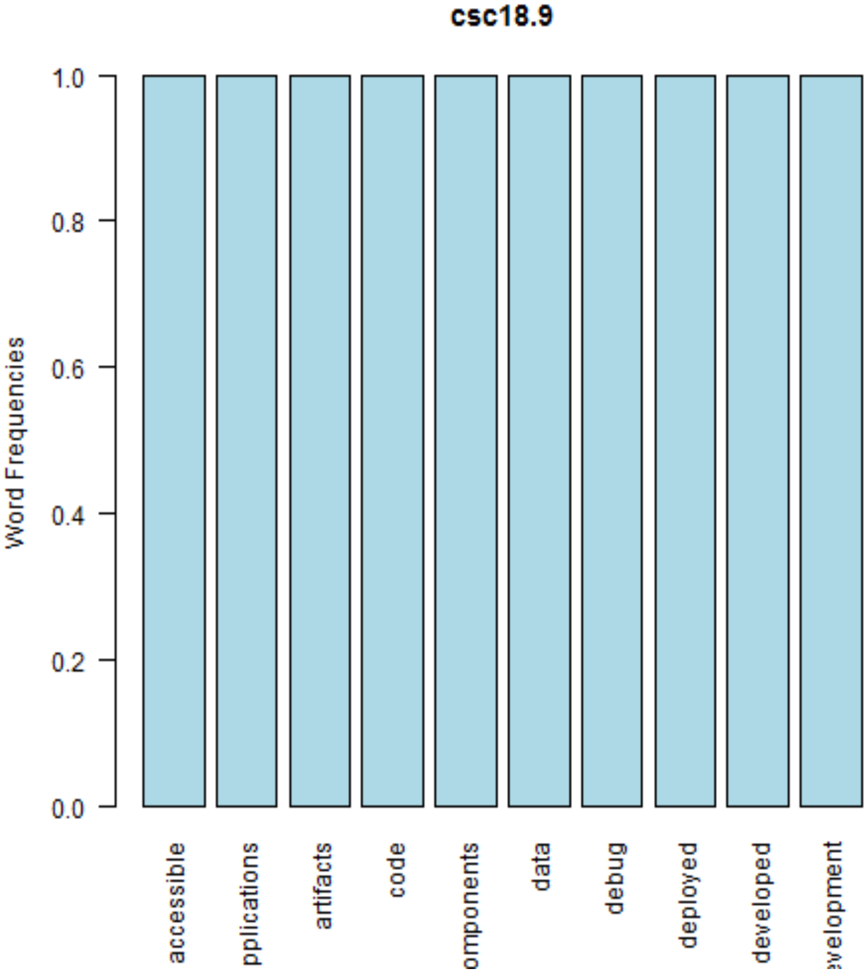
null device 1 [1] "Ensure that all software development personnel receive training in writing secure code for their specific development environment."

CSC 18.9

[1] “accessible + applications”



null device 1



null device 1 [1] “For in-house developed applications, ensure that development artifacts (sample data and scripts; unused libraries, components, debug code; or tools) are not included in the deployed software, or accessible in the production environment.”